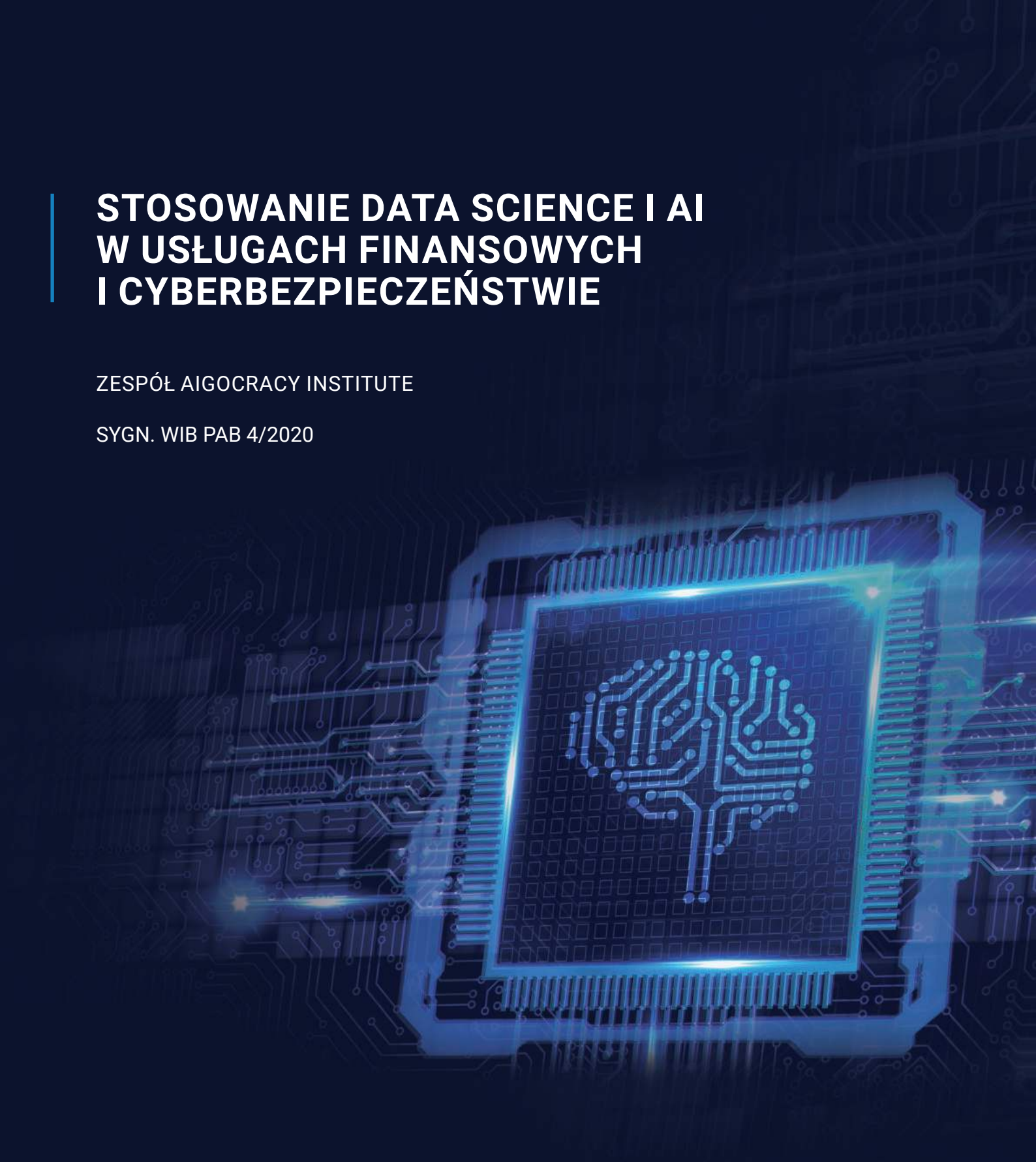
STOSOWANIE DATA SCIENCE I AI W USŁUGACH FINANSOWYCH I CYBERBEZPIECZEŃSTWIE

ZESPÓŁ AIGOCRACY INSTITUTE

SYGN. WIB PAB 4/2020



Raport opracowany na zlecenie Programu Analityczno-Badawczego
Fundacji Warszawski Instytut Bankowości

Poznań, 15.01.2020

PROGRAM
ANALITYCZNO
BADAWCZY

O Raporcie

Raport **Stosowanie Data Sclence i AI w usługach finansowych i cyberbezpieczeństwie** został opracowany na zlecenie Programu Analityczno-Badawczego Fundacji Warszawski Instytut Bankowości

Autorzy:

Raport przygotowany przez zespół Aigocracy Institute w składzie:

prof. Witold Abramowicz
dr Elżbieta Lewańska
dr Milena Stróżyna
dr Marcin Szmydt
Maciej Durski
Kacper Glazer
Mateusz Lemański
Oliwia Pietrzak
Ewelina Pożoga
Oskar Riewe-Perła
Joanna Wasilewska
Mateusz Wiatrowski

AIGOCRACY INSTITUTE
POWERED BY SYNERISE

Aigocracy Institute sp. z o.o. – instytut powstały w 2018 roku zorientowany jest na działania z obszaru Badań i Rozwoju (R&D), doradztwa oraz implementacji najnowszych rozwiązań w dziedzinie zaawansowanej analityki biznesowej, wykorzystującej rozwiązania z obszaru data science, sztucznej inteligencji oraz uczenia maszynowego. W zespole AGI znajdują się osoby posiadające wieloletnie doświadczenie w zróżnicowanych obszarach, takich jak biometria, wykorzystanie danych z IoT, analiza oraz prognozowanie ryzyka i anomalii, m.in. w transporcie morskim, technologiach semantycznych, profilowaniu behawioralnym, ekstrakcji danych z nieustrukturyzowanych źródeł, technologiach Big Data, algorytmach sztucznej inteligencji. Doświadczenie zdobyto zagranicą i w Polsce w trakcie realizacji szeregu projektów przemysłowych i badawczych. Aigocracy Institute zapewnia interdyscyplinarne oraz zindywidualizowane podejście do problemów biznesowych. Dzięki doświadczeniu pracowników AGI we współpracy z podmiotami z różnych branż, dysponują oni multidyscyplinarnym know-how i są w stanie w sposób holistyczny adresować problematykę analizy danych w zróżnicowanych zastosowaniach.

PROGRAM ANALITYCZNO BADAWCZY

O Programie

Program Analityczno-Badawczy przy Fundacji WIB powstał w 2019 roku jako odpowiedź na potrzeby sektora bankowego w zakresie analizy zjawisk, tworzenia opracowań i porządkowania wiedzy w obszarach cyberbezpieczeństwa i nowych technologii, a także szeroko rozumianego otoczenia sektora bankowego, kształtującego warunki działania banków w Polsce. Prace analityczno-badawcze w ramach programu realizowane są pod kątem możliwości praktycznego wykorzystania ich wyników w celu rozwoju sektora bankowego, podnoszenia poziomu bezpieczeństwa usług oraz kreowania wartości dla klientów bankowości. PAB WIB kładzie duży nacisk na rozwój współpracy ze środowiskami akademickimi i eksperckimi, poszukując synergii w zakresie zainteresowań badawczych autorów oraz potrzeb rozwojowych sektora bankowego.

W ramach programu realizowane są analizy i badania w następujących obszarach:

- **Nowe technologie i cyberbezpieczeństwo**
- **Zdolność banków do finansowania gospodarki**
- **Bankowość spółdzielcza**
- **Rynek nieruchomości**
- **Zielony ład i finansowanie energetyki odnawialnej**
- **Finansowanie projektów innowacyjnych**

Więcej na temat działalności programu na stronie www.pab.wib.org.pl

Kontakt:

dr Andrzej Banasiak
Koordynator Programu
m: 696 405 104
e: abanasiak@wib.org.pl

Jacek Gieorgica
m: 603 626 254
e: jgieorgica@wib.org.pl

Warszawski Instytut Bankowości
00-394 Warszawa, ul. Solec 38, lok 104
t: (22) 182 31 70
e: pab@wib.org.pl

Agnieszka Nierodka
m: 607 484 391
e: anierodka@wib.org.pl

dr Tomasz Pawlonka
m: 505 917 778
e: tpawlonka@wib.org.pl

Spis treści

Streszczenie kierownicze

Zawartość raportu i jego cele

Celem raportu jest przedstawienie algorytmów sztucznej inteligencji i uczenia maszynowego, które aktualnie są wykorzystywane dla celów zapewnienia cyberbezpieczeństwa oraz analiza potencjalnych możliwości przeniesienia rozwiązań opartych na AI do sektora bankowego. W dokumencie przedstawiono:

- przegląd metod uczenia maszynowego i uczenia głębokiego (Deep Learning),
- zagadnienie profilowania behawioralnego oraz pojęcie śladu cyfrowego,
- jedno z najistotniejszych zastosowań data science w cyberbezpieczeństwie, tj. metody wykrywania anomalii oraz działalności botów.

Wnioski

Data science jest szeroko stosowane w obszarze cyberbezpieczeństwa. Zarówno metody uczenia maszynowe, jak i uczenia głębokiego (Deep Learning) są obecnie dojrzałymi rozwiązaniami.

Uczenie maszynowe jest dziedziną, której celem jest wykorzystanie rozwiązań informatycznych i metod statystycznych do stworzenia systemu, który potrafiłby doskonalić swoje działania na podstawie tzw. danych uczących. System uczący się potrafi tworzyć nowe pojęcia i wnioskować na podstawie wiedzy fragmentarycznej. W raporcie przedstawiono metody uczenia maszynowego, z podziałem na uczenie nadzorowane (k-najbliższych sąsiadów, regresji liniowej, regresji logistycznej, drzewa decyzyjne, lasy losowe, naiwny klasyfikator Bayesa, SVMs, ensemble methods), uczenie nienadzorowane (analiza skupień, redukcja wymiarowości) oraz uczenie przez wzmacnianie (Q-learning, Deep Q Network). Metody uczenia nadzorowanego są stosowane w problemach klasyfikacji i regresji. W cyberbezpieczeństwie metody te służą do wykrywania złośliwego oprogramowania lub złośliwych adresów IP, jak również wykrywania niepożądanych działań (np. prób włamania). W usługach finansowych wykorzystuje się je do predykcji zajścia określonych zjawisk, np. podlegających ubezpieczeniu. Algorytmy uczenia nienadzorowanego są wykorzystywane przede wszystkim do segmentacji obiektów, np. klientów, zachowań. Pozwalają na identyfikację anomalii bez wcześniejszego uwzględnienia ich w zbiorze testowym.

Uczenie głębokie (ang. *Deep Learning*) polega na budowie sztucznych sieci neuronowych (ANN), tj. algorytmów mających na celu naśladowanie działania biologicznych sieci neuronowych. Systemy oparte na ANN na podstawie pewnego zbioru danych treningowych „uczą się" wykonywać określone zadania i automatycznie identyfikować cechy obiektów. Sztuczne sieci neuronowe są jednym z najbardziej popularnych kierunków rozwoju AI. Wykorzystuje się wiele odmian tych algorytmów, np. głębokie sieci neuronowe, konwolucyjne sieci neuronowe, rekurencyjne sieci neuronowe. W cyberbezpieczeństwie ANN są stosowane np. do wykrywania złośliwego oprogramowania, włamań, oszustw, prób wyłudzania informacji.

Niezależnie od stosowanych metod analitycznych, ich skuteczność jest uzależniona od wolumenu i jakości przetwarzanych danych. Analiza śladu cyfrowego, tj. informacji pozostawianych przez użytkownika na odwiedzanych przez niego witrynach internetowych, oraz informacji o zachowaniu użytkownika, stwarza nowe możliwości wykrywania anomalii (np. odróżniania ludzi od botów) oraz profilowania. Ślady cyfrowe są wykorzystywane np. do autoryzacji użytkownika, śledzenia ruchu klientów w sieci czy w analizie kredytowej. Profilowanie behawioralne pozwala na nadzór ruchu w sieci, umożliwia ochronę przed wcześniej trudnymi do wykrycia wyłudzeniami oraz automatyzację działań związanych z klasyfikacją użytkowników i wykrywaniu anomalii w ich zachowaniu.

Wykrywanie anomalii jest jednym z najważniejszych zadań związanych z cyberbezpieczeństwem. Jest stosowane np. do detekcji złośliwego oprogramowania, wykrywania włamań do sieci, wykrywania podejrzanego zachowania pracowników firmy,

wczesnego wykrywania zagrożeń. W wielu przypadkach w wykrywaniu anomalii wymagane jest przetwarzanie nie tylko statycznego zbioru danych, ale również strumieni danych (zbioru obserwacji zarejestrowanych w odstępach czasu). Szczególną uwagę poświęca się metodom wykrywania anomalii, które pozwalają zidentyfikować tzw. złe boty, czyli złośliwego oprogramowania, które symuluje zachowanie człowieka. Boty stanowią duże zagrożenie dla cyberbezpieczeństwa, ponieważ są wykorzystywane w przejmowaniu tożsamości, nielegalnym pobieraniu danych ze stron internetowych czy atakach typu DDoS.

Najważniejsze rekomendacje

- Skuteczność działania metod analizy danych jest uzależniona od dostępności zbioru danych uczących o odpowiednim wolumenie i jakości. Wiele podmiotów sektora bankowego, w szczególności małe banki, nie są w stanie zgromadzić zbioru pozwalającego na skuteczne wytrenowanie modeli analitycznych. Postulowana jest więc wymiana danych pomiędzy podmiotami, np. poprzez stworzenie wspólnego repozytorium danych uczących.
- Wygenerowanie przez pojedynczy podmiot szeregu czasowego danych treningowych i testowych może być długotrwałym procesem oraz nie obejmować wszystkich możliwych klas obiektów. Integracja zbiorów gromadzonych przez wiele podmiotów umożliwi szybsze trenowanie nowych modeli analitycznych.
- Ze względu na charakter danych gromadzonych w sektorze bankowym, budowa współdzielonego repozytorium danych lub systemów integrujących wiele podmiotów wymaga zachowania przyjętych norm bezpieczeństwa przetwarzania informacji (np. RODO, PN-ISO/IEC 20000-1, PN-ISO/IEC 20000-2) oraz wytycznych Uchwały nr 125 Rady Ministrów z dnia 22 października 2019 r. w sprawie Strategii Cyberbezpieczeństwa Rzeczypospolitej Polskiej na lata 2019–2024.
- Szczegóły dotyczące wspólnych rozwiązań analitycznych nacelowanych na poprawę cyberbezpieczeństwa całego sektora bankowego, w tym architektura współdzielonego repozytorium danych pochodzących od wielu podmiotów powinny zachować poufny charakter. Wymagania funkcjonalne i pozafunkcjonalne powinny zostać opracowane przy udziale przedstawicieli zainteresowanych podmiotów.

Słowa kluczowe

Uczenie maszynowe, sztuczna inteligencja, AI, uczenie głębokie, wykrywanie anomalii, cyberbezpieczeństwo, profilowanie behawioralne, ślad cyfrowy, analiza danych, eksploracja danych.

Wstęp

Sektor bankowy jest jednym z najbardziej wrażliwych i narażonych na cyberataki. Jak szacuje Międzynarodowy Fundusz Walutowy, średnie roczne straty poniesione przez instytucje finansowe na skutek cyberataków mogą sięgać 100 miliardów dolarów (stanowi to około 9% globalnych dochodów netto banków), potencjalnie zagrażając stabilności finansowej (International Monetary Fund, 2018). MFW przewiduje, że w niedalekiej przyszłości straty mogą wzrosnąć nawet 3-krotnie, sięgając 270-350 miliardów dolarów. Według Atlantic Council (Atlantic Council, 2018) gwałtownie rosną zagrożenia związane z cyberprzestępczością, a ich koszty w tej dekadzie przekroczą 1,5% globalnego PKB. Banki o mniejszych budżetach na ochronę przed cyberprzestępczością, do których zaliczają się banki w Polsce, narażone są na wyższą ekspozycję na działania przestępców w cyberprzestrzeni, niż w większości krajów rozwiniętych.

Podobne statystyki obserwowane są również w skali światowej. Między rokiem 2015 a 2018 liczba ofiar cyberprzestępczości w USA wzrosła z 13,1 miliona do 16,7 miliona (wzrost o 27%), zaś straty wzrosły z 15,5 mld USD do 16,8 mld USD. Obecnie tylko 3% cyberataków wykorzystuje luki technologiczne i w oprogramowaniu, zaś 97% stosuje technologie określane jako „social engineering" (Nicolls, 2018). Obserwowany jest też szybki wzrost zagrożeń związanych z impersonacją i kradzieżami tożsamości, np. straty poniesione na skutek przejmowania kont bankowych od roku 2016 do 2018 wzrosły o 120% (Pascual i in. 2018).

Technologie stosowane przez cyberprzestępców stają się coraz bardziej zaawansowane, wykorzystują AI i uczenie maszynowe, a dzięki usługom typu CaaS (Unni, 2018) (Cybercrime As A Service), obserwowany jest podział ról pomiędzy produkującymi narzędzia do popełniania przestępstw, a korzystającymi z tych narzędzi. Znika w ten sposób próg kompetencji technologicznych ograniczający przestępczość. Pojawiają się nowe modele fraudów, a celem ataków częściej stają się urządzenia mobilne. W konsekwencji, część społeczeństwa może utracić zaufanie do bankowości elektronicznej i obawia się korzystania z internetowych kanałów komunikacji.

Eksperci zajmujący się cyberbezpieczeństwem wskazują na szereg niepokojących zjawisk obserwowanych obecnie (European Payment Council, 2018):

- Cyberprzestępcy wykazują się coraz większym profesjonalizmem i skutecznością.
- Metody ataków przesuwają się od złośliwego oprogramowania w kierunku social engineering i kombinacji obu form ataku.
- W ramach social engineering coraz częściej celem ataków stają się osoby na wysokich stanowiskach (tzw. CEO fraud) oraz instytucje finansowe.
- Coraz częściej celem ataków stają się urządzenia mobilne, co jest silnie związane ze wzrostem popularności bankowości mobilnej oraz brakiem doświadczenia wielu nowych użytkowników tejże bankowości. Większość banków działających w Polsce podejmuje działania zachęcające do większej intensywności korzystania z bankowości mobilnej. W konsekwencji, ekspozycja na opisywane ryzyko będzie znacząco rosła także w Polsce.
- Pojawiła się usługa CaaS (Cybercrime As A Service), związana z szybko rosnącym poziomem automatyzacji cyberprzestępczości.
- Wzrost cyberzagrożeń w sektorze finansowym jest także związany z presją na poprawę „user -experience" aplikacji internetowych i mobilnych (co często jest w konflikcie z bezpieczeństwem) oraz z wdrożeniem Dyrektywy PSD2, która otwiera płatniczy łańcuch wartości na nowe wyzwania, takie jak brak wystarczająco efektywnej autentykacji klientów przez firmy e-commerce inicjujących płatność przez bank.

W tym kontekście zapewnienie cyberbezpieczeństwa jest kluczowe dla stabilności gospodarczej kraju i musi być zapewnione zarówno w skali makro (na poziomie całego sektora bankowego), jak i mikro (z punktu widzenia klienta). Kluczowa jest tutaj analiza danych na poziomie całego sektora bankowego, która umożliwiłaby wczesne wykrywanie anomalii i opracowanie systemu ostrzegania w czasie rzeczywistym wszystkich podmiotów z sektora o pojawiających się zagrożeniach. Wolumen danych, który musi być przetwarzany w celu wykrycia szeroko zakrojonych ataków, jest zbyt duży dla tradycyjnych metod. Ponadto, większość metod AI wymaga dużego wolumenu danych uczących, pozwalających na wytrenowanie modeli i odkrycie reguł wnioskowania. W konsekwencji, dane zgromadzone przez jeden podmiot mogą być niewystarczające do wykrycia anomalii, np. w postaci próby nieuprawnionego logowania. Konieczne może okazać się wykorzystanie metod profilowania behawioralnego i analizy śladów cyfrowych pozostawianych przez rzeczywistych użytkowników oraz boty w czasie interakcji z witrynami i aplikacjami różnych podmiotów (banków).

Rozwój rozwiązań w zakresie AI powoduje, że narzędzia wykorzystujące metody z tego obszaru są coraz szerzej stosowane przez przestępców (tzw. „złe AI"), np. do tworzenia botów. Z drugiej strony,

sektor bankowy dysponuje coraz większym wolumenem danych o zachowaniu użytkowników lub przeprowadzonych transakcjach. Zastosowanie metod pozwalających na automatyczne przetwarzanie tych danych oraz, co ważniejsze, automatycznie wnioskowanie, pozwoli na wykrywanie działania „złego AI" oraz wczesną neutralizację zagrożeń.

Celem raportu jest przedstawienie algorytmów sztucznej inteligencji i uczenia maszynowego, które aktualnie są wykorzystywane dla celów zapewnienia cyberbezpieczeństwa oraz analiza potencjalnych możliwości przeniesienia rozwiązań opartych na AI do sektora bankowego.

Struktura raportu jest następująca. Rozdział 2 Algorytmy Machine Learning i ich zastosowanie w cyberbezpieczeństwie przedstawia najpopularniejsze algorytmy uczenia maszynowego oraz ich zastosowanie w usługach finansowych i cyberbezpieczeństwie. Rozdział 3 poświęcony jest metodom Deep Learning, w tym sztucznym sieciom neuronowym. Zagadnienie profilowania behawioralnego oraz śladów cyfrowych zostało omówione w rozdziale 4. Szczególnie istotne dla zapewnienia cyberbezpieczeństwa jest automatyczne wykrywanie anomalii, w związku z czym temu zagadnieniu poświęcono rozdział 5. Raport kończą rekomendacje zastosowania poszczególnych metod w wybranych obszarach (rozdział 6).

Algorytmy Machine Learning i ich zastosowanie w cyberbezpieczeństwie

Uczenie maszynowe jest dziedziną, której celem jest wykorzystanie rozwiązań informatycznych i metod statystycznych do stworzenia systemu, który potrafiłby doskonalić swoje działania na podstawie tzw. danych uczących. System uczący się potrafi tworzyć nowe pojęcia i wnioskować na podstawie wiedzy fragmentarycznej. Uczenie maszynowe rozwijane jest od lat 50. XX wieku i obecne stanowi jeden z głównych obszarów w AI.

2.1. Kryteria podziału uczenia maszynowego

Według kryterium nadzoru można wyszczególnić następujące typy algorytmów uczenia maszynowego, które mogą być stosowane w cyberbezpieczeństwie:

- **Nadzorowane:** Uczenie nadzorowane wykorzystuje dane uczące z rozwiązaniami (etykietami). Jest wykorzystywane w klasyfikacji i regresji. Klasyfikacja jest to przydzielenie obserwacji do prawidłowej kategorii. Regresja to natomiast proces, podczas którego ustala się związek pomiędzy zmiennymi, dostarcza informacji o tym, czy są to zmienne zależne, czy niezależne. Polega na predykcji wartości numerycznej (Ayodele, 2010).

- **Nienadzorowane:** Uczenie nienadzorowane wykorzystuje dane, które nie są oznakowane (nie mają etykiet). Jest wykorzystywane w analizie skupień, która polega na wyszczególnianiu względnie jednolitych grup elementów. Innym przykładem użycia uczenia nienadzorowanego jest redukcja wymiarowości, której celem jest uproszczenie danych bez utraty wyekstrahowanych informacji. Cel ten jest osiągany poprzez ekstrakcję cech, czyli scalenie kilku skorelowanych ze sobą cech w jedną (Ayodele, 2010).

- **Półnadzorowane:** Uczenie półnadzorowane wykorzystuje dane częściowo oznakowane, czyli tylko niektóre przykłady zawierają etykiety. Na tym algorytmie bazują między innymi głębokie sieci przekonań.

- **Uczenie przez wzmacnianie:** Uczenie przez wzmacnianie polega na nauce najlepszej strategii działania, nazywanej polityką. System uczący, nazywany agentem, uczy się poprzez odbieranie kar i nagród. Polityka definiuje rodzaj działania jakie powinno być wybrane w danej sytuacji, tak aby, uzyskiwać największą nagrodę.

Kryterium możliwości uczenia w czasie rzeczywistym pozwala wyróżnić:

- **Metody uczenia wsadowego:** Model jest trenowany na zbiorze danych, który nie może być wzbogacany o nowe rekordy. Po modyfikacji zbioru treningowego, konieczne jest ponowne wytrenowanie modelu.

- **Metody uczenia przyrostowego:** Model jest trenowany na bieżąco poprzez sekwencyjne dostarczanie danych. Takie podejście sprawdza się w sytuacjach, gdy program wymaga szybkiego dostosowywania się do nowych warunków, czyli ma wysoki współczynnik uczenia. Jednocześnie szybko zapomina on o poprzednich danych, czyli ich waga maleje.

2.2. Uczenie nadzorowane

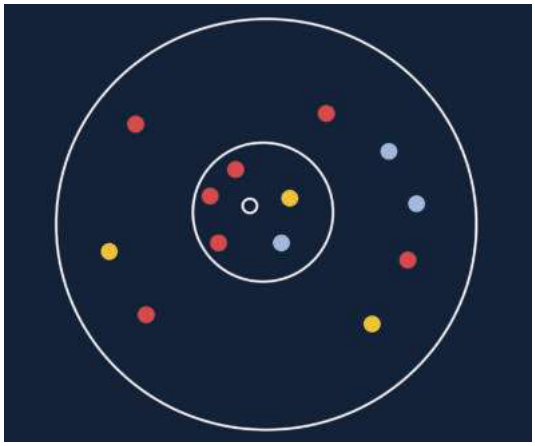
Metody uczenia nadzorowanego wymagają udziału człowieka w procesie tworzenia funkcji odwzorowującej systemu (tj. przekształcającej dane wejściowe do zadanego formatu wyjściowego). Nadzór sprowadza się do dostarczenia zestawu danych uczących.

W kolejnych sekcjach przedstawione zostały przykłady algorytmów uczenia nadzorowanego, które są lub mogą być stosowane w cyberbezpieczeństwie i/lub bankowości.

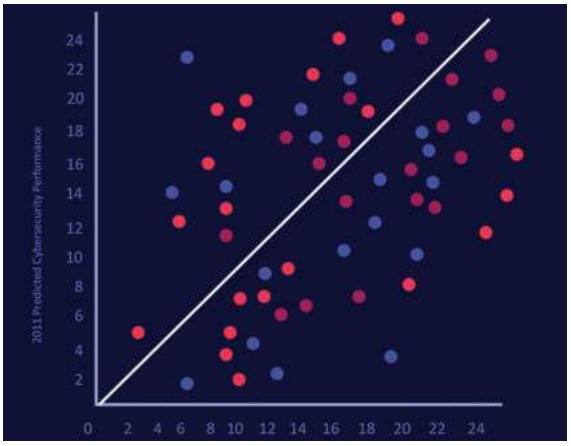
2.2.1. Metoda k-najbliższych sąsiadów

Metoda k-najbliższych sąsiadów, nazywana również klasyfikatorem k-NN (ang. *k-nearest neighbours*), służy do grupowania danych. Predykcja przynależności nowego obiektu (obserwacji) do klasy bazuje na porównaniu ze zbiorem uczącym, składającym się z obiektów, z których każdy opisany jest przez wektor cech oraz etykietę (klasę). Do określenia odległości pomiędzy obiektami najczęściej używana jest miara odległości Euklidesa (Owsiak i Owsiak, 2010). Inne przykładowe miary odległości to np. odległość kątowa, współczynnik korelacji liniowej Pearsona, miara Gowera, odległość Manhattan, Canberra, Czybyszewa (Wierzchoń i in., 2015). Wybór miary odległości zależy od typu analizowanych danych (binarnych, nominalnych, numerycznych) (Bartłomowicz i in. 2011).

Algorytm klasyfikacji k-NN może zostać wykorzystany do wykrywania nieprawidłowości, na przykład włamań do systemu, ponieważ pozwala na klasyfikację zachowania programu jako „normalne" lub „podejrzane" na podstawie sekwencji wywołań systemowych, wykorzystanych do scharakteryzowania programu (Liao i Vemuri, 2002).



Rysunek 1. Metoda k-najbliższych sąsiadów. W otoczeniu nowego obiektu znajdują się trzy obiekty czerwone, jeden niebieski i jeden żółty. Nowy obiekt zostanie więc zaklasyfikowany do klasy „czerwone".
Źródło: opracowanie własne.



Rysunek 2. Regresja liniowa.
Źródło: Linking Cybersecurity Policy and Performance (Kleiner I in. 2014).

2.2.2. Regresja liniowa

Regresja liniowa pozwala na szacowanie wartości danych na podstawie innych danych ze zbioru. Analiza regresji jest bardzo popularną i chętnie stosowaną techniką, pozwalającą opisywać związki zachodzące pomiędzy zmiennymi wejściowymi (objaśniającymi) a wyjściowymi (objaśnianymi). Przewidywanie wartości zmiennych objaśnianych na podstawie wartości zmiennych objaśniających jest możliwe dzięki znalezieniu tzw. modelu regresji (Faraway, 2002).

Regresja liniowa znajduje zastosowanie podczas budowania modelu prognostycznego, wyjaśniającego rozbieżności pomiędzy faktyczną a przewidywaną wydajnością w zakresie cyberbezpieczeństwa. Rysunek 2 przedstawia rozproszenie rzeczywistego i przewidywanego poziomu bezpieczeństwa. Wyniki znajdujące się powyżej linii są uważane za lepsze od przewidywanych. Oznacza to, że ich faktyczny poziom bezpieczeństwa jest lepszy niż prognozuje model. Analogicznie, punkty znajdujące się poniżej linii nie osiągają zadowalających wyników, a rzeczywiste poziomy bezpieczeństwa cybernetycznego są gorsze niż przewidywał model (Kleiner I in. 2014).

2.2.3. Regresja logistyczna

Regresja logistyczna to funkcja określająca prawdopodobieństwo przynależności do jednej z dwóch grup, czyli klasyfikacja binarna. Wartości zmiennej objaśnianej wskazują zatem na wystąpienie lub brak wystąpienia prognozowanego zdarzenia. Algorytm ten może być wykorzystywany w przypadku zależności innych niż liniowe (Pociecha i in., 2014).

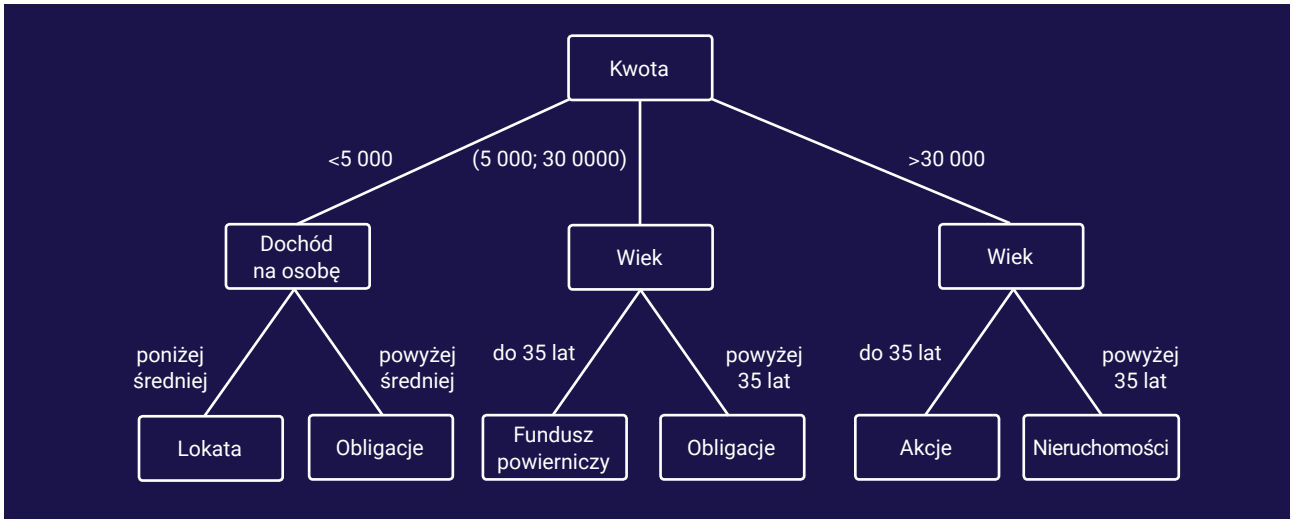
Aktuariusze ubezpieczeń na życie wykorzystują regresję logistyczną do przewidywania, w oparciu o dane dotyczące ubezpieczającego (np. wiek, płeć, wynik badania fizykalnego), szansy, że ubezpieczający umrze przed upływem terminu obowiązywania polisy.

2.2.4. Drzewa decyzyjne

Drzewo decyzyjne to metoda wspomagająca procesy decyzyjne. Każdy węzeł drzewa reprezentuje cechę (atrybut), łącze (gałąź) reprezentuje decyzję (regułę), a każdy liść reprezentuje wynik. Można wyróżnić drzewa klasyfikacyjne, w których odpowiedź jest wartością binarną (należy lub nie należy do danej klasy) oraz drzewa regresyjne, w których przewidywany wynik jest liczbą rzeczywistą. Drzewa decyzyjne mogą reprezentować dowolnie złożone pojęcia pojedyncze lub wielokrotne, jeżeli tylko ich definicje da się wyrazić jako wartości atrybutów. W metodzie drzew decyzyjnych testuje się wartość jednego atrybutu na raz, co powoduje niepotrzebny rozrost drzewa dla danych, gdzie poszczególne atrybuty zależą od siebie (Singh Rathore, Kumar, 2016) (Yadav i in., 2016).

Tworzenie drzewa decyzyjnego pozwala stosunkowo łatwo i szybko stworzyć bazę reguł dla rozwiązywanego problemu. Każdy węzeł takiego drzewa reprezentuje albo pytanie o wartość atrybutu, albo konkluzję. Każda gałąź wychodząca z węzła związanego z pytaniem o atrybut reprezentuje jedną z możliwych wartości tego atrybutu. Przykładowe atrybuty i ich możliwe wartości w domenie usług finansowych, przedstawione na Rysunku 4, to:

- Inwestycje: lokata w banku, obligacje, fundusz powierniczy, akcje, nieruchomości.



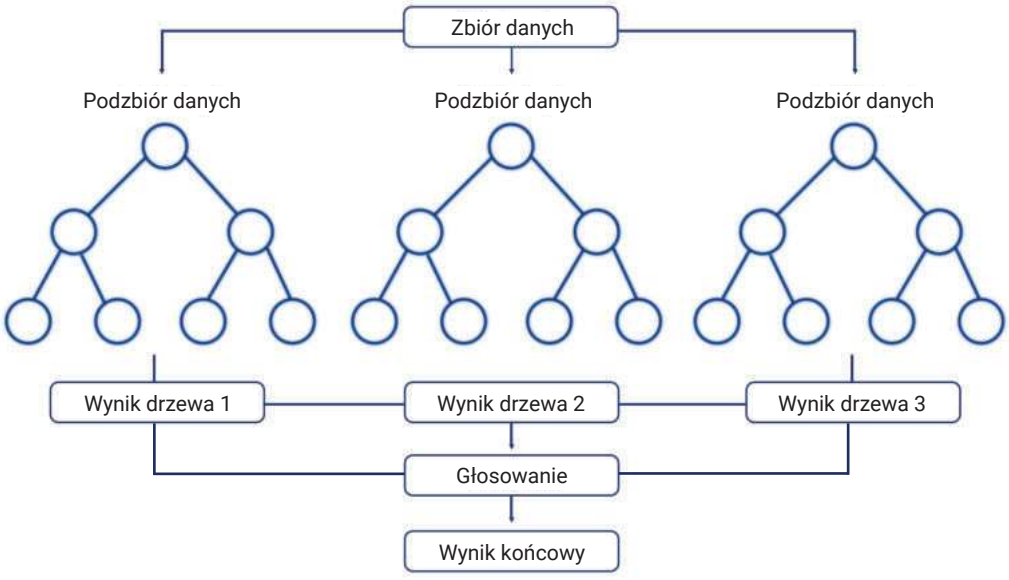
Rysunek 3. Lasy losowe.
Źródło: opracowanie własne.

- Kwota do zainwestowania: do 5000, od 5000 do 30 000, powyżej 30 000.
- Wiek inwestora: do 35 lat, powyżej 35 lat.
- Dochód na osobę: poniżej średniej krajowej, powyżej średniej krajowej.

Drzewa decyzyjne znajdują również zastosowanie w wykrywaniu włamań do sieci. Dane z poprawnie działającej sieci (bezpiecznej) są wykorzystywane do stworzenia reguł drzewa decyzyjnego. Decyzje podejmowane na węzłach pozwalają na wykrywanie anomalii, które mogą być wywoływane próbą nieautoryzowanego dostępu do sieci (włamania do sieci) (Markey i Atlasis, 2011).

2.2.5. Lasy losowe

Model lasu losowego (ang. *random forest*) to zbiór modeli drzew decyzyjnych, które są łączone w celu tworzenia prognoz. Tworząc las losowy należy określić liczbę drzew decyzyjnych, które będą składały się na model (Rysunek 3). Algorytm lasu losowego pobiera losowo próbki obserwacji z danych treningowych i buduje model drzewa decyzyjnego dla każdej próbki. Losowe próbki są zwykle pobierane ze zwracaniem, co oznacza, że tę samą obserwację można pobrać wiele razy. Końcowym rezultatem jest zbiór drzew decyzyjnych, które są tworzone z różnymi grupami obserwacji danych



Rysunek 4. Drzewa decyzyjne.
Źródło: http://home.agh.edu.pl/~pmarynow/pliki/iwmet/drzewa.pdf.

pochodzących z oryginalnych danych treningowych. Każde z drzew zwraca decyzję lub krotkę prawdopodobieństw klasyfikacji. Decyzje (prawdopodobieństwa) z drzew wchodzących w skład lasu są traktowane jako głosy, a wynikiem prognozy jest ta decyzja, która otrzymała najwięcej głosów (której średnie prawdopodobieństwo było najwyższe). Głosy mogą być ważone lub nie, w zależności od wybranej przez użytkownika metody głosowania. Lasy losowe mogą być stosowane zarówno do klasyfikacji, jak i regresji. Istotną cechą tej metody jest również fakt, że im więcej drzew decyzyjnych zostanie wykorzystanych, tym bardziej trafne prognozy można otrzymać (Vincenzi i in., 2011).

2.2.6. Naiwny klasyfikator Bayesowski

Naiwny klasyfikator Bayesowski jest to prosty klasyfikator probabilistyczny oparty na twierdzeniu Bayesa i założeniu o niezależności zmiennych (dlatego też jest nazywany modelem cech niezależnych). Algorytm ten szczególnie nadaje się do problemów o wielowymiarowych danych wejściowych (Patil i Sherekar, 2013).

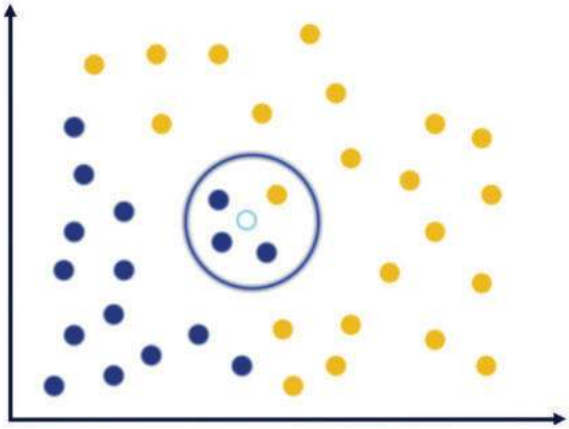
Przykładowy problem klasyfikacji zaprezentowany jest na Rysunku 5 i dotyczy przydzielenia obiektu do klasy niebieskiej lub pomarańczowej. Liczba elementów niebieskich wynosi 20, a pomarańczowych 40. Zatem prawdopodobieństwo a priori (określane przed doświadczeniem) przynależności obiektu do poszczególnych klas wynosi odpowiednio:

- dla obiektów niebieskich = liczba obiektów niebieskich w sąsiedztwie nowego elementu / całkowita liczba niebieskich obiektów (3/20)
- dla obiektów pomarańczowych = liczba obiektów pomarańczowych w sąsiedztwie nowego elementu / całkowita liczba pomarańczowych obiektów (1/40).

Prawdopodobieństwo a posteriori, określające przynależność do poszczególnej grupy:

- dla obiektów pomarańczowych = prawdopodobieństwo a priori * prawdopodobieństwo hipotezy (wynikające z liczby obiektów w kolorze pomarańczowym) (1/40)
- dla obiektów niebieskich = prawdopodobieństwo a priori * prawdopodobieństwo hipotezy (wynikające z liczby obiektów w kolorze niebieskim) (1/60)

Nowy obiekt zostanie zaklasyfikowany jako pomarańczowy, ponieważ przynależność do tej grupy ma wyższe prawdopodobieństwo a posteriori.



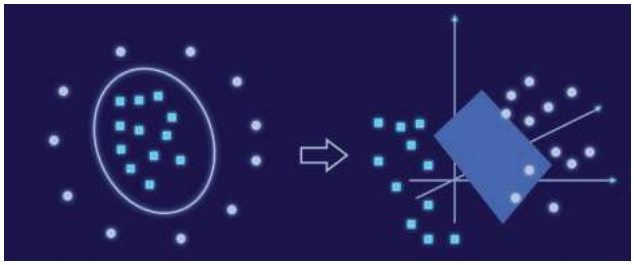
Rysunek 5. Naiwny klasyfikator Bayesowski.
Źródło: opracowanie własne.

Naiwny klasyfikator Bayesowski znajduje zastosowanie w wykrywaniu włamań. Wówczas, dla każdej wartości atrybutu wyliczane jest prawdopodobieństwo, że jest to zagrożenie i na tej podstawie klasyfikowane są nowe obiekty (Panda i Patra, 2007).

2.2.7. SVM – support vector machines

SVM to algorytm najczęściej wykorzystywany do klasyfikacji, ale znajduje zastosowanie również w badaniu regresji. Jest to klasyfikator dyskryminujący, czyli wyznaczający granicę między skupieniami danych. Dane przedstawiane są w przestrzeni n-wymiarowej, gdzie n jest liczbą elementów wektora. Wartości poszczególnych elementów wektora są jednocześnie wartościami współrzędnych w przestrzeni n-wymiarowej. Klasyfikacja polega na znalezieniu hiperpłaszczyzny, która rozdziela oba skupienia (hiperpłaszczyzna ma wymiar o 1 mniejszy niż przestrzeń, w której się znajduje). W przypadku gdy problem jest nieliniowy, można zastosować *kernel function*, polegającą na przeniesieniu problemu do wyższego wymiaru, rozwiązaniu go, a następnie zwróceniu wyniku (Brown i in. 2000).

Przykład wykorzystania *kernel function* prezentuje Rysunek 6: przestrzeń dwuwymiarowa z nieliniowym problemem klasyfikacji jest przenoszona do przestrzeni



Rysunek 6. Kernel function.
Źródło: opracowanie własne.

trójwymiarowej, gdzie różnokolorowe punkty są rozdzielane na grupy za pomocą hiperpłaszczyzny.

W cyberbezpieczeństwie metoda SVM jest wykorzystywana do klasyfikacji obiektów, na przykład do określania czy adres strony (URL) jest złośliwy, czy bezpieczny (Halder i Ozdemir, 2018).

2.2.8. Ensemble methods

Ensemble methods (metody zespołowe, metody grupowania) służą do poprawy wyników uczenia maszynowego poprzez łączenie kilku modeli. Celem zastosowania metod grupowania jest zmniejszenie wariancji, czyli określenie, jak duże jest zróżnicowanie wyników, czy są one skoncentrowane wokół średniej (*bagging*), zmniejszenie odchylenia, czyli różnicy pomiędzy szacowanymi wartościami a rzeczywistymi (*boosting*) lub poprawy prognoz (*stacking*) (Dietterich, 2000). Ważnymi pojęciami związanymi z metodami zespołowymi są:

- *weak learners* – tzw. słabi uczniowie – modele, które są trenowane do rozwiązywania zadanego problemu niezależnie od pozostałych modeli (inaczej modele bazowe).
- *homogenous weak learners* – wykorzystuje jednorodną bazę modeli, trenowanych tą samą metodą (zwykle stosowane przy *bagging* i *boosting*).
- *heterogenous weak learners* – wykorzystuje bazę modeli trenowanych różnymi metodami (zwykle stosowane przy *stacking*).

Ważne jest, aby wybór podstawowych algorytmów był spójny z metodą agregacji algorytmów, czyli jeżeli wykorzystywane są modele o niskiej wariancji i dużym odchyleniu standardowym, metoda agregacji powinna mieć tendencję do zmniejszania odchylenia. Istnieje kilka sposobów agregacji modeli. W przypadku regresji, dane wyjściowe poszczególnych modeli można uśrednić, aby uzyskać dane wejściowe modelu zespolonego. W przypadku klasyfikacji, klasa wyprowadzona przez każdy model może być postrzegana jako głos, a klasa, która otrzyma większość głosów, jest zwracana przez model zespołowy (nazywa się to twardym głosowaniem). Można również wziąć pod uwagę prawdopodobieństwa zwrócenia każdej klasy przez poszczególne modele, uśrednić je i utrzymać klasę z najwyższym średnim prawdopodobieństwem (nazywa się to miękkim głosowaniem). Średnie lub głosy mogą być proste lub ważone, jeśli można zastosować odpowiednie wagi.

Do łączenia słabych uczniów wykorzystywane są meta-algorytmy, wśród których wymienić można:

- *Bagging – najczęściej używany przy jednorodnych modelach bazowych, które są uczone równolegle i niezależnie od siebie, a następnie są łączone. Bagging opiera się na metodzie statystycznej nazywanej bootstraping, która służy do szacowania parametrów zbioru danych, poprzez uśrednianie wyników małych próbek, które są generowane z tego zbioru (*Seni i Elder, 2010). Metoda bagging opiera się na dopasowaniu kilku niezależnych modeli i uśrednieniu ich wyników w celu otrzymania modelu o mniejszej wariancji (Rysunek 7). W praktyce, ze względu na niewystarczającą ilość danych, trudno jest otrzymać w pełni niezależne modele, dlatego stosuje się próbki bootstrap[1], aby uzyskać modele prawie niezależne. Bagging może być stosowany zarówno przy problemie klasyfikacji, jak i *regresji.*

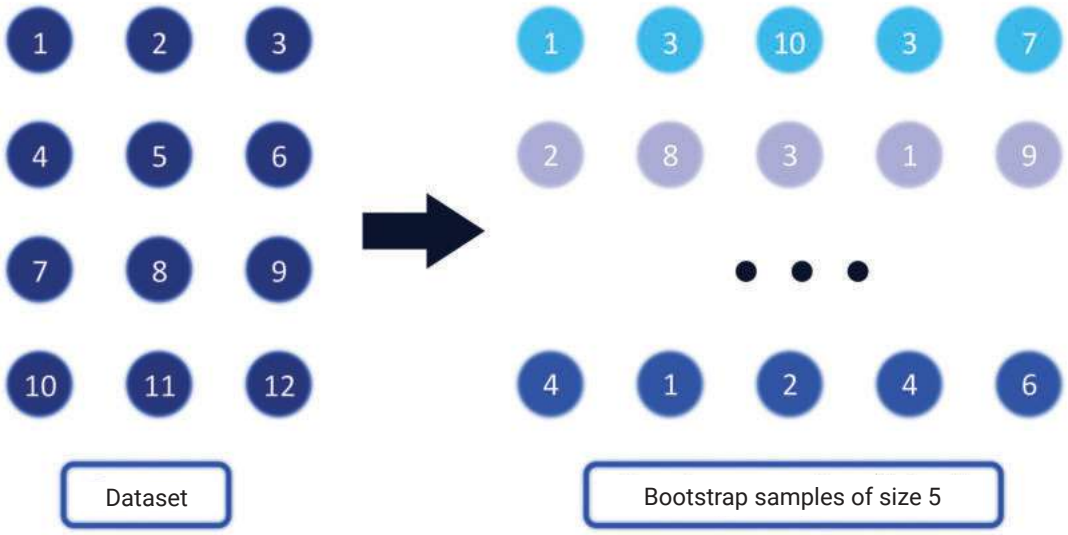
- *Boosting – najczęściej używany przy jednorodnych modelach bazowych, które są uczone sekwencyjnie. Polega na dopasowywaniu wielu słabych uczniów. Każdy model w sekwencji jest dopasowywany, co nadaje większą wagę obserwacjom, które były niewłaściwie sklasyfikowane przez poprzednie modele w sekwencji (*Drucker i in. 1994). Istnieją dwie najważniejsze metody dostosowania modelu na podstawie modeli bazowych (Seni i Elder, 2010): *adaptive boosting* oraz *gradient boosting. Boosting* może być wykorzystywany zarówno w zadaniu klasyfikacji, jak i regresji.

- *Stacking* – często wykorzystuje niejednorodne modele bazowe, które uczone są równolegle. Wynik jest zwracany przez meta-model jako prognoza bazująca na prognozach otrzymanych z modeli bazowych.

2.3. Uczenie nienadzorowane

Uczenie nienadzorowane wykorzystuje algorytmy, które szukają wzorców w nieoznaczonych danych (tj. nie jest przypisana do nich żadna wartość klasyfikująca) (Sathya, Abraham, 1986). Uczenie nienadzorowane jest wykorzystywane między innymi w analizie skupień (rozdz. 2.3.1 Algorytmy analizy skupień), metodach redukcji wymiarowości (rozdz. 2.3.2 Algorytmy redukcji wymiarowości) oraz wykrywaniu anomalii (Rozdział 5 Wykrywanie anomalii). Wybierając algorytm analizy skupień, należy wziąć pod uwagę czy możliwe jest zdefiniowanie docelowej liczby skupień przed rozpoczęciem analizy – algorytmy niehierarchiczne wymagają takiego parametru. W przypadku dużej liczby analizowanych obiektów i małej liczby skupień, algorytmy niehierarchiczne charakteryzują się mniejszą złożonością obliczeniową, niż hierarchiczne.

[1] Próbka bootstrap – „próbka z próbki", tworzone poprzez wybór elementów z próbki generalnej, a następnie zwrot tych elementów do próbki generalnej i powtórzenie tego procesu tyle razy, ile próbek chce się uzyskać.



Rysunek 7. Bagging.
Źródło: opracowanie własne.

Algorytmy analizy skupień są wykorzystywane w segmentacji klientów, wyszukiwaniu nietypowych zachowań (np. złośliwych aktywności). Brak wymaganego zbioru uczącego pozwala również na wykrywanie przy ich pomocy nowych, wcześniej niewystępujących ataków.

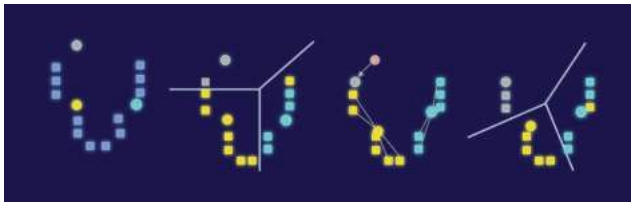
2.3.1. Algorytmy analizy skupień

Analiza skupień (ang. *clustering) polega na analizie zbioru danych w celu zidentyfikowania w nim grup (skupień) podobnych obiektów, tj. takich, które zawierają obiekty podobne bardziej do siebie, niż do obiektów z innych skupień (*Rokach i Maimon, 2005). Klasteryzacja wykorzystywana jest w wielu różnych dziedzinach, przykładem może być segmentacja klientów w celu dostarczenia spersonalizowanej oferty lub identyfikacja poszczególnych typów nowotworów w medycynie (Tan i in., 2006). Znaczenie i złożoność zagadnień związanych z analizą skupień przyczyniły się do stworzenia wielu odmian algorytmów tego typu.

Podobnie jak w metodzie k-NN (patrz. 2.2.1 Metoda k-najbliższych sąsiadów), istotne znaczenie dla efektywności algorytmów analizy skupień ma wybór miary odległości (im większa odległość obiektów, tym mniej są one do siebie podobne).

Metody niehierarchicznej analizy skupień

Najpopularniejszym algorytmem niehierarchicznej analizy skupień jest algorytm k-średnich. Wymaga on określenia z góry liczby grup (skupień), na które ma zostać podzielony zbiór danych. Następnie dla każdej grupy identyfikuje się punkt nazywany centroidem. Początkowo centroidy można wyznaczyć losowo wśród obiektów w zbiorze. W kolejnym kroku jest obliczany dystans między obiektami, a każdym z centroidów. Następnie obiekty są przypisywane do grupy, w której znajduje się najbliższy im centroid. Kolejno wyznaczane są nowe centroidy – jedną z metod jest obliczenie średniej arytmetycznej wszystkich punktów należących do grupy. Następnie zostaje powtórzony krok obliczenia odległości i ponownego przypisania obiektów do grup. Iteracje powtarzamy założoną liczbę razy lub do momentu, w którym punkty w grupach nie zmieniają swojej przynależności po wyznaczeniu nowej pozycji centroidów (Tan i in, 2006) (Li i Wu, 2012) (Riad i in., 2013). Przykład działania algorytmu k-średnich został zaprezentowany na Rysunku 8.

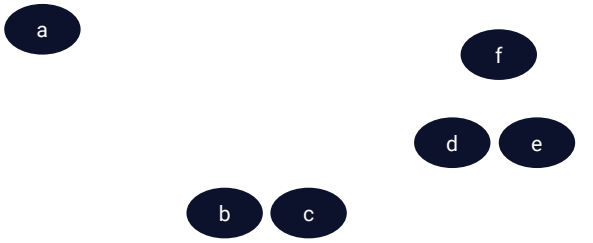


Rysunek 8. K-średnich – kolejne kroki algorytmu.
https://en.wikipedia.org/wiki/K-means_clustering

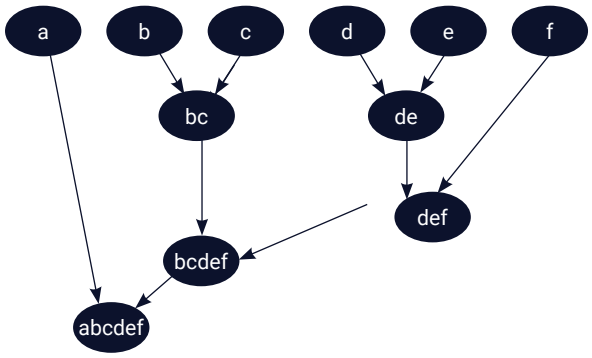
Metody hierarchicznej analizy skupień

W początkowym stanie metody aglomeracyjnej każdy obiekt stanowi osobne, jednoelementowe skupienie. W kolejnych iteracjach skupienia znajdujące się najbliżej siebie (tj. najbardziej do siebie podobne) są łączone. Rysunek 9 prezentuje surowy zbiór danych, a Rysunek 10 kolejne iteracje, w których obiekty są

grupowane, aż do momentu stworzenia jednego zbioru. W tej metodzie wykorzystywane są takie algorytmy, jak reprezentatywny (ang. *representative algorithm), czy algorytm połączeń (ang. links algorithm)* (Abdullah, Hamdan, 2015).



Rysunek 9. Niepogrupowany zbiór danych.
Źródło: opracowanie własne.



Rysunek 10. Iteracje grupowania zbioru danych.
Źródło: opracowanie własne.

Metody podziałowe

Metody podziałowe działają odwrotnie do metod aglomeracyjnych, tj. w stanie początkowym wszystkie obiekty należą do tego samego skupienia. W kolejnych iteracjach są one dzielone na mniejsze skupienia. Metoda jest kontynuowana do momentu, gdy każdy obiekt będzie stanowił osobne skupienie lub do momentu, w którym zostanie spełniony warunek ukończenia (Reddy i in., 2017). Dla tej metody wykorzystywane są takie algorytmy jak PDDP (ang. *Principal Direction Divisive Partitioning*) lub k-średnich (Savaresi i in., 2002).

Metody analizy skupień oparte na gęstości

Analiza skupień oparta na gęstości (ang. *Density-based clustering*) polega na identyfikacji skupień na podstawie bliskości obiektów w przestrzeni. W metodzie nie jest wydzielane centrum zbioru. Kolejne obiekty są dodawane do grupy na podstawie ich odległości między sobą – identyfikacja dwóch grup zachodzi, gdy w przestrzeni danych są one oddzielone polem o niskim zagęszczeniu obiektów. Pola te są jednocześnie traktowane jako szumy lub wartości odstające i nie są wliczane do żadnej z grup (Kriegel i in. 2011). Algorytmy oparte na gęstości są często wykorzystywane do analizy danych geolokalizacyjnych i obrazów. Dla tej metody bardzo popularnym algorytmem jest DBSCAN (ang. *Density-Based Spatial Clustering of Applications with Noise*) (Moreira I in. 2005).

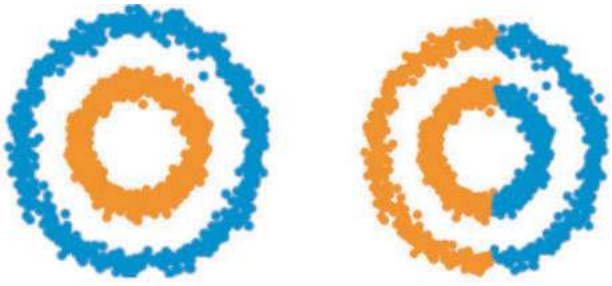
Metody rozmytej analizy skupień

Metody klasyfikacji służą do grupowania elementów na podstawie ich podobieństwa. Wcześniej omawiane metody dotyczyły tzw. twardej klasyfikacji (ang. *hard clustering*), czyli dzieliły zbiór obiektów na klastry rozłączne, tj. takie, w których obiekt może być przypisany tylko do jednej grupy. Metody rozmytej analizy skupień (ang. *soft clustering, fuzzy clustering*) tworzą grupy w taki sposób, że obiekt ze zbioru danych może przynależeć do kilku skupień. Podział zachodzi za pomocą wskazania poziomu członkostwa, który wskazuje stopień, z jakim obiekt należy do różnych grup. Jednym z popularniejszych algorytmów w miękkiej klasyfikacji jest algorytm c-średnich (ang. *fuzzy c-Means*) (Grover, 2014).

Porównanie algorytmów analizy skupień

Poszczególne metody analizy skupień przeznaczone są do różnych zastosowań. Każdorazowo metoda powinna zostać dostosowana do wybranego zbioru danych, funkcji celu oraz możliwości obliczeniowych.

Zastosowanie różnych algorytmów grupowania na tym samym zbiorze danych może zwrócić zupełnie inne wyniki. Przykładowo, Rysunek 11 prezentuje wynik działania algorytmów k-średnich i DBSCAN. Analiza skupień na podstawie gęstości (lewy rysunek) pozwoliła na zidentyfikowanie dwóch grup, w których obiekty znajdujące się blisko względem siebie trafiły do jednej grupy. Algorytm k-średnich (prawy rysunek), stworzył dwa centroidy, do których zostały przydzielone obiekty.



Rysunek 11. Rezultat algorytmu DBSCAN (lewy) oraz K-średnich (prawy).
Źródło: https://scikit-learn.org/stable/_images/sphx_glr_plot_cluster_comparison_001.png

Wybierając algorytm analizy skupień, należy wziąć pod uwagę czy możliwe jest zdefiniowanie docelowej liczby skupień przed rozpoczęciem analizy – algorytmy niehierarchiczne wymagają takiego parametru. W przypadku dużej liczby analizowanych obiektów i małej liczby skupień, algorytmy niehierarchiczne charakteryzują się mniejszą złożonością obliczeniową, niż hierarchiczne.

Algorytmy analizy skupień są wykorzystywane w segmentacji klientów, wyszukiwaniu nietypowych zachowań (np. złośliwych aktywności). Brak wymaganego zbioru uczącego pozwala również na wykrywanie przy ich pomocy nowych, wcześniej niewystępujących ataków.

2.3.2. Algorytmy redukcji wymiarowości

- Niezależnie od metody, każdy zbiór danych poddawany analizie składa się z obiektów, które opisywane są przez zbiór cech (np. wielkość, kolor, ciężar, wartość, etc.). Każda taka cecha to wymiar, w którego przestrzeni opisywany jest obiekt. Liczba wymiarów wpływa na złożoność modelu, dlatego redukcja wymiarowości ma na celu jego uproszczenie (Vlachos, 2010). Przykładowo, wyszukując anomalie w transakcjach, analizowane obiekty mogą być opisane cechami, które dla wykrycia anomalii nie są istotne (np. data zaksięgowania transakcji przez bank). Większa liczba cech wymaga znacznie większej ilości danych, aby pokryć możliwe kombinacje cech tworzące wzorzec. W związku z tym, występuje większe ryzyko przetrenowania modelu, co znaczy, że model będzie działać poprawnie jedynie na danych treningowych, przy słabej skuteczności na danych rzeczywistych. Dodatkowo wymagana jest większa moc obliczeniowa, pamięć na dysku oraz wydłużony jest czas trenowania. Im większa wielowymiarowość, tym trudniej również wizualizować zbiór danych.

Wyróżnia się dwie techniki redukcji wymiarowości: selekcja i ekstrakcja cech.

Selekcja cech

Selekcja cech opiera się na pozostawieniu najważniejszych cech i odrzuceniu pozostałych. Metody selekcji zostały podzielone na trzy kategorie: (Jovic i in., 2015) (Venkatesh i Anuradha, 2019)

- Filtrowanie – w tej kategorii model wybiera najlepszy podzbiór cech w oparciu o miary statystyczne, takie jak korelacja Pearsona, liniowa analiza dyskryminacyjna, ANOVA i in.

- Metoda pakowania (ang. *Wrapper method*) – w tej metodzie podzbiory cech są wybierane w oparciu o algorytmy indukcyjne. Po wybraniu podzbioru szacowana jest dokładność modelu, a następnie, w zależności od dokładności modelu z poprzedniej iteracji, podejmowana jest decyzja o zwiększeniu lub zmniejszeniu zbioru cech. Ta metoda jest wolniejsza niż metoda filtrowania, jednak, jak zostało dowiedzione empirycznie, jest bardziej skuteczna (Jovic i in., 2015). W tej metodzie używa się takich algorytmów, jak liniowy SVM (patrz 2.1.7 SVM – support vector machines) lub naiwny klasyfikator bayesowski (patrz 2.2.6 Naiwny klasyfikator Bayesowski).

- Metoda osadzona (ang. *Embedded methods*) – metoda ta jest wbudowana i zintegrowana jako część algorytmu uczenia się. Jednym z popularniejszych algorytmów w tej metodzie jest drzewo decyzyjne.

Ekstrakcja cech

Ekstrakcja cech (ang. *Feature Extraction*) redukuje wielowymiarowość zbioru przez łączenie kilku oryginalnych cech w jedną. Celem jest stworzenie nowego zbioru danych, przy jak najmniejszej zmianie jego struktury oraz wariancji (Vlachos, 2010). Wśród algorytmów służących do ekstrakcji cech można wyróżnić:

- Analiza Głównych Elementów Składowych (ang. *Principal Components Analysis*, PCA) – jedna z najbardziej popularnych metod w redukcji wymiarowości. Jest to technika liniowa, która oznacza, że wykonuje się redukcję wymiarowości przez osadzenie danych w liniowej podprzestrzeni o niższej wartości wymiarowości (Van Der Maaten i in., 2007).

- Liniowe Osadzenie Lokalne (ang. *Locally Linear Embeddings*, LLE) – „wykorzystuje mapowanie liniowe w celu uchwycenia lokalnych relacji sąsiedzkich reprezentatywnych dla lokalnej geometrii kolektora danych, które są następnie w miarę możliwości zachowane w powiązanym, niskowymiarowym zbiorze punktów" (Carreira-Perpinan, 2001).

- t-dystrybuowane Stochastyczne Osadzanie Sąsiada (ang. *t-distributed Stochastic Nieghbor Embedding*, t-SNE) – „wizualizuje dane wielowymiarowe, nadając każdemu punktowi lokalizację na mapie dwuwymiarowej lub trójwymiarowej" (Maaten i Hinton, 2008).

2.4. Uczenie przez wzmacnianie

Uczenie przez wzmacnianie nie wymaga zbioru zaklasyfikowanych danych uczących. Agenci (programy) trenowani są poprzez nagrody i kary będące wynikiem

ich oddziaływania na otoczenie. Pozwala tym samym na odkrycie przez system strategii działania, która pozwoli maksymalizować nagrody.

Metody z tej grupy pozwalają na opracowanie skalowalnych i adaptujących się do otoczenia systemów cyberbezpieczeństwa (Nguyen i Reddi, 2019).

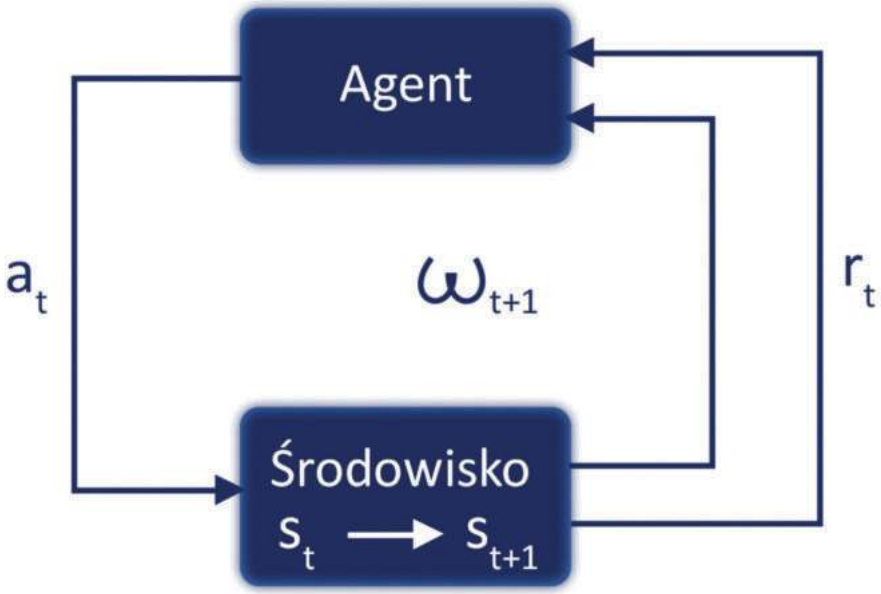
Uczenie przez wzmacnianie (ang. *reinforcement learning*, RL) jest obszarem z zakresu uczenia maszynowego, w którym system sztucznej inteligencji jest przedstawiony w formie agenta, oddziałującego na środowisko. Agent jest w stanie wykonywać działania w swoim otoczeniu, ale także je obserwować. Środowisko przekazywane agentowi jest w formie czarnej skrzynki, czyli nie ma przedstawionych żadnych założeń dotyczących modelu i jego struktury. Dzięki temu możliwe jest zastosowanie RL bez konieczności wprowadzania specyficznych zmian. Agent wie, jakie działania ma podjąć, ale musi sam odnaleźć te, które przynoszą największe korzyści (tzw. nagrody). Tym samym możemy zaobserwować, że RL jest w stanie radzić sobie z problemem sekwencyjnego podejmowania decyzji. Agent uczy się dobrego zachowania na podstawie prób i eksperymentów, nie wykorzystując przy tym doświadczenia zdobytego a priori. Dane dostępne są w kolejnych sekwencjach i stopniowo wykorzystywane do aktualizacji zachowania agenta. Agent jest w stanie zbierać informacje na temat „najciekawszej" części środowiska, co sprawia, że podejścia RL zapewnią lepszą wydajność obliczeniową (Sutton i Barto, 2018) (Szepesvari, 2010) (Francois-Lavet i in. 2018) (Koziarski i in. 2018).

Schemat RL przedstawiony jest na Rysunku 12, gdzie agent rozpoczyna w danym stanie w środowisku $s_0 \in S$, w celu zebrania wstępnych obserwacji $w_\omega \in \Omega$. W każdym momencie czasu, agent musi podjąć akcję $a_t \in A$. Następstwem tych działań jest: uzyskanie przez agenta nagrody $r_t \in R$, przejście do stanu $s_{t+1} \in S$ oraz uzyskanie obserwacji $w_{\omega+1} \in \Omega$ przez agenta. Nagroda definiuje, które wydarzenia są dobre, a które nie (Francois-Lavet i in. 2018).

W celu lepszego zrozumienia działania RL, przedstawmy prosty przykład, gdzie agent monitoruje poziom naładowania baterii urządzenia i wysyła sygnały do modułu kontroli. Środowiskiem agenta jest w tym przypadku reszta urządzenia oraz jego otoczenie. Podobny przykład dotyczy bezpieczeństwa w banku, gdzie agent monitoruje poprawność logowania, a reszta czynników (np. zachowania, ślad cyfrowy) to jego środowisko.

RL różni się od uczenia nadzorowanego, gdyż rozwiązuje problem uczenia się z interakcji i własnego doświadczenia, a nie zbioru danych uczących. Podobnie jak metody uczenia nienadzorowanego, uczenie przez wzmacnianie wykorzystuje nieoznakowane zbiory, lecz w przypadku uczenia nienadzorowanego celem jest odnalezienie określonej struktury, zaś w RL poprzez system nagród, następuje wzmocnienie procesu uczenia się. Dlatego też uczenie przez wzmocnienie jest uważane za trzeci paradygmat uczenia się maszynowego.

Algorytmy uczenia przez wzmacnianie najczęściej klasyfikuje się na trzy grupy: oparte na wartościach, oparte



Rysunek 12. Oddziaływanie agenta ze środowiskiem w RL.
Źródło: opracowanie własne na podstawie (Francois-Lavet i in. 2018)

na politykach i oparte na modelach. Klasa algorytmów opartych na wartościach ma na celu zbudowanie funkcji wartości, która pozwoli na zdefiniowanie polityki. Przykładem takiego podejścia jest algorytm QLearning (Watkins i Dayan, 1992). Metody oparte na politykach, to szczególna sekcja algorytmów wykorzystujących metody gradientowe (np. Actor-Critic Method (Konda i Tsitsiklis, 2000) lub Natural Policy Gradients (Jiang i in. 2015)). Metody te optymalizują cel wydajności, czyli oczekiwaną skumulowaną nagrodę, poprzez znalezienie odpowiedniej polityki. Ostatnim rodzajem jest podejście oparte na modelach. Skupia się ono na modelu środowiska (funkcja dynamiki i narody) w połączeniu z algorytmem planowania (np. Pure model-based), co pozwala na rozważenie wielu różnych możliwych sytuacji, nawet jeżeli jest jedna sekwencja działania (Francois-Lavet i in. 2018).

Algorytmy wykorzystujące uczenie przez wzmacnianie, które znajdują swoje zastosowanie w tematach związanych z cyberbezpieczeństwem w różnych dziedzinach nauki:

- **Q-learning** – jest jedną z technik wykorzystywanych przy uczeniu przez wzmacnianie. Według Watkinsa (Watkins i Dayan, 1992) algorytm ma na celu, na podstawie skończonego zestawu stanów i akcji dla jednego agenta, wybrać optymalną strategię. Następuje to za pomocą uczenia się funkcji wartości akcji, czyli przez sekwencyjne modyfikowanie wartości funkcji zgodnie z kolejnymi obserwacjami. Wprowadzona zostaje funkcja Q, która ma za zadanie zwracać nagrodę świadczącą o „jakości" akcji w danym stanie.

Powyższy algorytm można przedstawić w następujący sposób:

$$Q^n(s_t, a_t) \leftarrow (1 - \alpha) \cdot Q(s_t, a_t + \alpha \cdot (r_t + \gamma \cdot maxQ(s_{t+1}, a))$$

gdzie Q^n to nowa wartość funkcji, a_t to akcja, s_t to stan, γ to czynnik dyskontowy (ang. *discount factor*), który determinuje ważność przyszłych nagród, α to współczynnik uczenia (ang. *learning rate*), który przyjmuje wartości z przedziału (0, 1) (Watkins i Dayan, 1992) (Matiisen, 2015) (Manju i Punithavalli, 2011).

- **Deep Q Network (DQN)** – jest to Q-Learning w połączeniu z sieciami neuronowymi. Jest to sieć wielowarstwowa, radząca sobie z niedociągnięciami Q-Learning, np. problemem korelacji między kolejnymi obserwacjami. DQN losowo odtwarza próbki poprzednich obserwacji przechowywane w pamięci, w celu dalszego treningu. Problem niestabilności w propagacji wstecznej jest także rozwiązywany poprzez zredukowanie wartości nagrody do zakresu [–1,1], zapobiegając w ten sposób nadmiernemu wzrostowi wartości Q (Behzadan i Munir, 2017). Algorytm został wprowadzony przez Mnihet i początkowo wykorzystany w środowisku gier ATAR[2], ucząc się bezpośrednio z pikseli (Mnih i in. 2015).

2.5. Zastosowanie uczenia maszynowego

Metody uczenia nadzorowanego są stosowane w problemach klasyfikacji i regresji. W cyberbezpieczeństwie metody te służą do wykrywania złośliwego oprogramowania lub złośliwych adresów IP, jak również wykrywania niepożądanych działań (np. prób włamania). W usługach finansowych wykorzystuje się je do predykcji zajścia określonych zjawisk, np. podlegających ubezpieczeniu.

Zestawienie przykładowych zastosowań metod uczenia nadzorowanego przedstawia Tabela 1.

[2] Przedsiębiorstwo branży informatycznej, specjalizujące się w tworzeniu różnego rodzaju gier zręcznościowych.

Tabela 1. Zastosowanie metod uczenia nadzorowanego w cyberbezpieczeństwie i usługach finansowych.

Nazwa metody	Opis zastosowania	Referencje
KNN	Klasyfikowanie zachowania programu jako normalne lub złośliwe	(Liao i Vemuri, 2002)
	Wykorzystywane przez algorytm LOF (Local Outlier Factor) do obliczania odległości między obserwacjami	(John i Naaz, 2019)
regresja liniowa	Ocena wydajności modelu predykcyjnego	(Kleiner I in. 2014)
	Oparte na danych podejście do konstruowania taksonomii zbiorów danych z badań nad bezpieczeństwem cybernetycznym	(Zheng i in., 2018)
regresja logistyczna	Wykrywanie i kategoryzacja złośliwego oprogramowania	(Mahdavifar i Ghorbani, 2019)
	Weryfikowanie czy roszczenie względem ubezpieczyciela jest fałszywe	(Mahdavifar i Ghorbani, 2019)
drzewa decyzyjne i lasy losowe	Generalny opis zastosowań, wykrywanie złośliwej działalności, wykrywanie nowych ataków, nadawanie priorytetów w IDS i określanie ich przyczyn źródłowych	(J. H. Wilson, 2009)
	Wykrywanie włamań	(Manish i in. 2012)
	Wykrywanie, klasyfikacja i charakterystyka złośliwego oprogramowania Android	(Markey i Atlasis, 2011)
naiwny klasyfikator Bayesowski	Wykrywanie włamań do sieci	(Li i in. 2017)
	Wykrywanie i kategoryzacja złośliwego oprogramowania	(Shahadat i in., 2017)
SVM	Klasyfikacja bezpiecznych i podejrzanych adresów IP	(Dangi, 2012)
	Wykrywanie i kategoryzacja złośliwego oprogramowania	(Mahdavifar i Ghorbani, 2019)
ensemble methods	Klasyfikacja mająca na celu polepszenie dopasowania marketingu produktów ubezpieczeniowych	(Lin i in. 2017)
	Predykcja bankructwa	(Mihalovic, 2016)
	Poprawa wyniku wykrywania oszustw ubezpieczeniowych	(Xu i in., 2011)

Źródło: opracowanie własne.

Algorytmy uczenia nienadzorowanego są wykorzystywane przede wszystkim do segmentacji obiektów, np. klientów, zachowań. Pozwalają na identyfikację anomalii bez wcześniejszego uwzględnienia ich w zbiorze testowym. Przykładowe zastosowanie przedstawiono w Tabeli 2.

Tabela 2. Zastosowanie metod uczenia nienadzorowanego w cyberbezpieczeństwie i usługach finansowych.

Nazwa metody	Opis zastosowania	Referencje
Algorytmy analizy skupień	Klasyfikowanie nowych, wcześniej niezdefiniowanych cyberataków	(Wei I in., 2018)
	Wykrywanie złośliwej aktywności i naruszeń polityki	(Ranjan i Sahoo, 2014)
	Segmentacja klientów bankowych	(Wu i Po-Hsuan Chou, 2011)
	Analizowanie logów pochodzących z serwera w celu wykrycia anomalii	(Riadi I in., 2013)
Algorytmy redukcji wymiarowości	Wstępne przetworzenie zbioru danych dla uzyskania lepszych wyników w późniejszych analizach	(Pajouh i in. 2016)

Źródło: opracowanie własne.

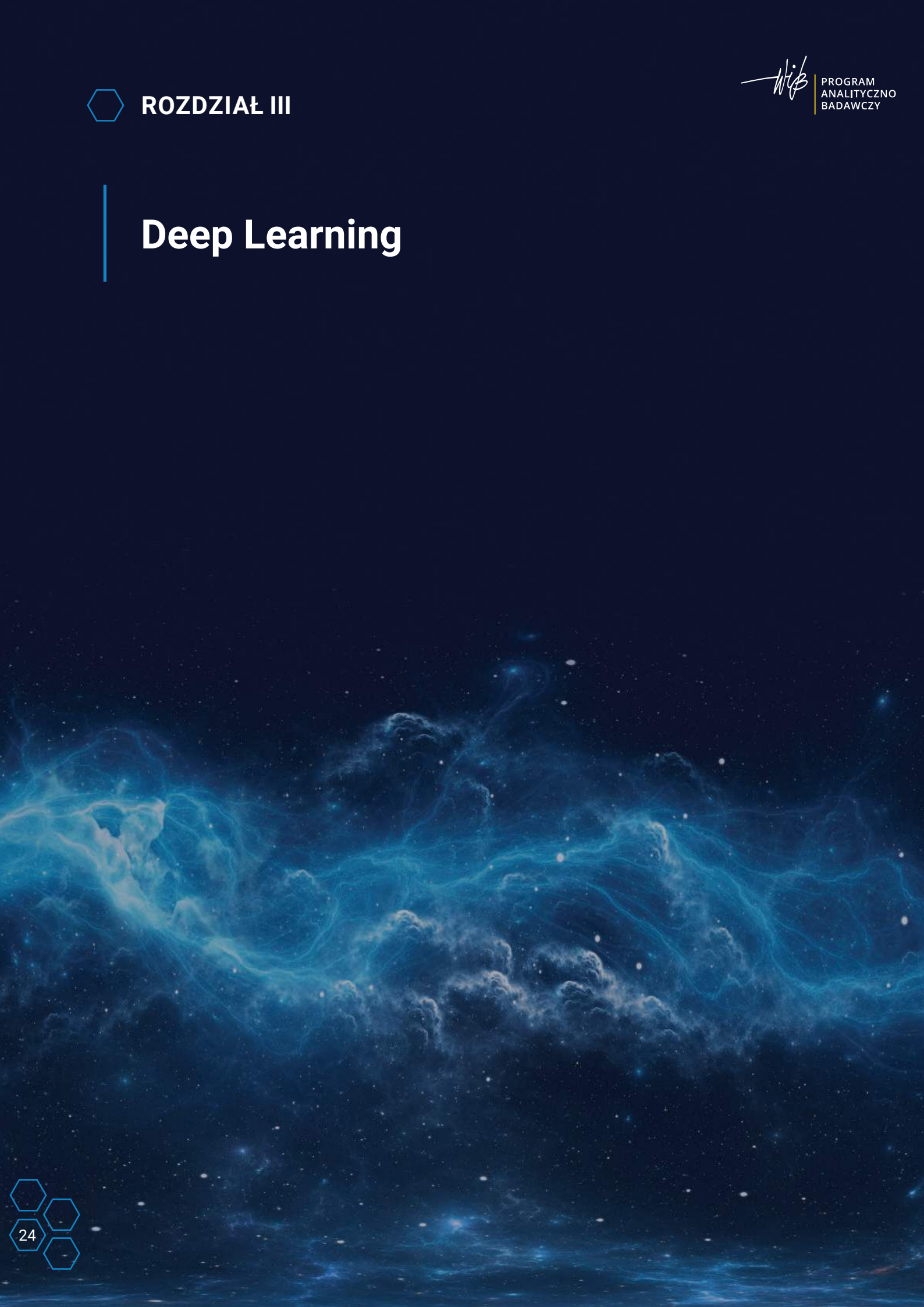
Przykładowe zastosowanie uczenia przez wzmacnianie zostało przedstawione w Tabeli 3.

Tabela 3. Zastosowanie metod uczenia przez wzmacnianie w cyberbezpieczeństwie i bankowości.

Nazwa metody	Opis zastosowania	Referencje
Q-Learning	Cyberbezpieczeństwo – mechanizm reakcji sztucznej inteligencji dla systemu IDS (Intrusion Detection Systems)	(Stefanova i Ramachandran, 2018)
	Bankowość – detekcja oszustw finansowych	(El Bouchti i in. 2017)
	Bankowość – zarządzanie środkami pieniężnymi w bankowości detalicznej	(State I in.,2019)
	Cyberbezpieczeństwo – analiza behawioralna	(Li i Qiu, 2019)
	Cyberbezpieczeństwo – wykrywanie włamań w systemie	(Nguyen i Reddi, 2019)
	Cyberbezpieczeństwo – wykrywanie włamań	(Gulmez, 2019), (Kim i Park, 2019)
	Cyberbezpieczeństwo – obrona przed atakami	(T. Chen i in. 2019
	Bankowość – prognozowanie ruchu cen akcji na giełdzie	(Dang, 2019)

Źródło: opracowanie własne.

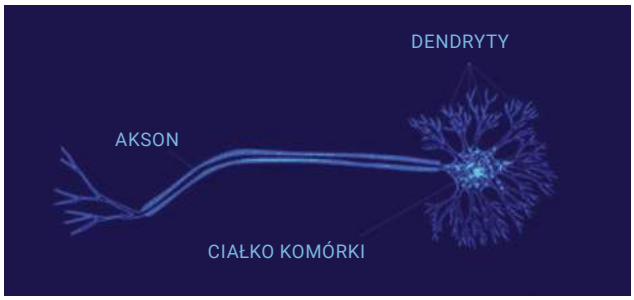
Deep Learning



Uczenie głębokie (ang. *Deep Learning*) jest jedną z technik sztucznej inteligencji, która naśladuje działanie ludzkiego mózgu w przetwarzaniu danych. Wykorzystywane jest głównie przy wspomaganiu podejmowania decyzji.

Głębokie uczenie znacznie zyskało na popularności w ostatniej dekadzie. Było to spowodowane między innymi szybkim rozwojem tej dziedziny nauki i coraz większą wydajnością różnych odmian sztucznych sieci neuronowych (Chandy i in., 2018).

Sztuczne sieci neuronowe (ang. *Artificial Neural Nnetworks*, ANN) to algorytmy mające na celu naśladowanie działania biologicznych sieci neuronowych. Systemy oparte na ANN „uczą się" wykonywać zadania i automatycznie generują cechy identyfikujące obiekty, biorąc przy tym pod uwagę przetwarzane przez siebie przykłady (Kriesel, 2007). Rozpoznawane przez nie wzorce są wektorami liczbowymi, reprezentującymi różne typy danych (np. dźwięk, tekst, obraz, szeregi czasowe). W niniejszym rozdziale omówiono sztuczny odpowiednik biologicznych sieci neuronowych, a także najważniejsze rodzaje modeli i algorytmy uczenia wykorzystywane w tych modelach.



Rysunek 13. Przybliżony model neuronu.
Źródło: (Stęgowski,2004)

Biologiczna sieć neuronowa składa się z miliardów komórek, które mają bardzo zbliżoną strukturę i ogromną liczbę połączeń między sobą. Naukowym terminem takiej komórki jest neuron. Terminologia taka obowiązuje także w przypadku ANN. Każdy neuron to komórka przetwarzająca informacje (Rysunek 11). Neuron złożony jest z ciała komórki oraz dwóch rodzajów rozciągających się gałęzi: dendrytów i aksonu. Neuron otrzymuje sygnały od innych neuronów przez dendryty i przekazuje te sygnały generowane przez ciało komórkowe wzdłuż aksonu. Funkcjonowanie neuronu można porównać zatem do działania przełącznika, neuron albo przekazuje sygnał dalej, jeżeli spełnione zostały jakieś warunki, albo pozostaje bezczynny. Funkcjonalną jednostką pomiędzy neuronami jest synapsa, której efektywność może być regulowana poprzez przechodzące przez nią sygnały, tak aby mogła „uczyć się" na podstawie aktywności, w których uczestniczy (Jain i in., 1996). Dzięki takiemu działaniu neuronów oraz dużego poziomu interaktywności między nimi, powstaje wiele możliwości przetwarzania informacji za pomocą sieci neuronowych.

3.1. Sztuczna sieć neuronowa (Artificial Neural Network)

Zarówno biologiczna, jak i sztuczna sieć neuronowa, jest bardzo złożonym systemem, umożliwiającym przetwarzanie dużych ilości informacji jednocześnie. Rozwój sztucznych sieci neuronowych prowadzi do naśladowania działania ludzkiego mózgu. W tej chwili w wielu zadaniach mózg ludzki jest bardziej wydajny niż komputer. Koncepcja sztucznej sieci neuronowej ma zatem umożliwić maszynom wykonywanie zadań na poziomie biologicznych sieci neuronowych.

Pierwsze podejścia do stworzenia ANN pojawiły się w 1943 roku, kiedy Warren McCulloch i Water Pitts stworzyli model obliczeniowy dla sieci neuronowych (McCulloch i Pitts, 1943). W roku 1949 D.O. Hebb stworzył pierwszą z zasad działania sieci neuronowych, która opierała się na tym, że jeżeli dwa neurony są jednocześnie aktywne, to waga pomiędzy nimi powinna być zwiększona (Hebb, 1949). W kolejnych latach, wraz z ogromnym rozwojem komputerów, teoria ANN poczyniła znaczne postępy. Wiele badań koncentrowało się wówczas na sieciach z ukrytą jedną warstwą, trenowanych za pomocą *propagacji wstecznej* (Rumelhart i in, 1986). Obecnie sztuczne sieci neuronowe znajdują zastosowanie w wielu dziedzinach nauki oraz przemysłu (Ganesan i in. 2010) (Ripley, 2007) (Yin i in., 2015), tj. są wykorzystywane do klasyfikacji, grupowania, prognozowania, czy optymalizacji.

Sieci neuronowe posiadają następujące cechy i właściwości (Haykin, 2010):

- **Nieliniowość** – sztuczny neuron może być nieliniowy. Połączone ze sobą nieliniowe neurony tworzą nieliniową sieć. Jest to bardzo ważna cecha ze względu na wprowadzane dane wejściowe, które mogą przyjmować nieliniowy charakter.

- **Mapowanie wejść-wyjść** – neurony są połączone ze sobą wagami, zatem każda zmiana danych wejściowych oznacza zmianę siły połączeń między neuronami. Wagi są tak modyfikowane, aby zminimalizować różnice między sygnałem wejściowym a wyjściowym. Sieć uczy się na przykładach konstruując mapowanie wejścia-wyjścia dla danego problemu.

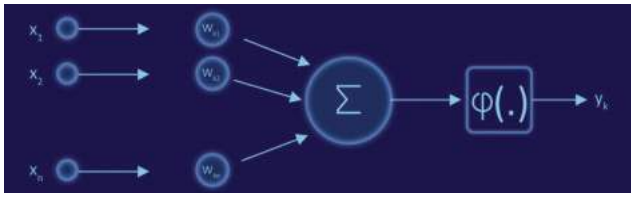
- **Adaptacyjność** – sieci są przystosowane do zmian w otaczającym je środowisku. Posiadają wbudowane funkcje umożliwiające korektę wag, w zależności od zmian i warunków działania, potrafiąc przy tym ignorować zakłócenia.

- **Reakcja z dowodami** – podczas klasyfikacji, sieć neuronowa może być tak zaprojektowana, aby dostarczyć informacji o wyborze wzoru i informacji o poziomie zaufania na temat podjętej decyzji. Pozwoli to np. na odrzucenie fałszywych wzorców i poprawę jakości sieci.

- **Informacje z kontekstu** – dzięki połączeniom, na każdy neuron wpływa informacja z innego neuronu, co powoduje, że sieć posiada informacje kontekstowe.

3.1.1. Budowa sieci neuronowych

Sztuczna sieć neuronowa składa się ze skierowanych, ważonych połączeń oraz z warstw, które podobnie jak w sieci biologicznej, zawierają proste jednostki przetwarzające – *neurony*. Siła połączenia jest nazywana *wagą* (Kriesel, 2007). Warstwa wejściowa reprezentuje dane wejściowe, które są wektorami składającymi się z wartości liczbowych. Mogą to być zarówno dane ustrukturyzowane (temperatura, cena itp.), jak i dane nieustrukturyzowane (piksele obrazu, tekst w języku naturalnym itp.). Podstawowa architektura sieci składa się z trzech warstw neuronowych: wejścia, warstwy ukrytej oraz wyjścia. W prostych sieciach neuronowych przepływ sygnału odbywa się od wejścia do wyjścia w jednym kierunku. Sieć powinna być tak zbudowana, aby umieszczony w niej zestaw danych stał się zamierzonym wynikiem. Wagi na początku są ustalane losowo, następnie w sieci osadzane są dane, na podstawie których wagi są korygowane zgodnie z zasadami uczenia. Szkolenie sieci można podzielić na uczenie nadzorowane, nienadzorowanie i uczenie przez wzmocnienie (patrz. 2.1 Kryteria podziału uczenia maszynowego) (Abraham, 2005).



Rysunek 14. Model sztucznego neuronu.
Źródło: opracowanie własne na podstawie (Haykin, 2010)

Rysunek 14 przedstawia model sztucznego neuronu. Zestaw danych wejściowych trafia do bloku wag synaptycznych, gdzie każdy sygnał x_i z synapsy i połączonej z neuronem k jest przemnożony odpowiednio przez wagę w_{ki}. Należy zwrócić uwagę, że w przeciwieństwie do wag synaps w mózgu, waga sztucznego neuronu może być zarówno wartością negatywną, jak i pozytywną. Następnie suma iloczynu wag i danych wejściowych staje się argumentem funkcji aktywacji φ.

Model sztucznego neuronu można także przedstawić za pomocą wzorów matematycznych. Sygnał z danych wejściowych $x_1, x_2, ..., x_n$ jest jednokierunkowy, podobnie jak przepływ sygnału wyjściowego, i zostaje poddany liniowej modyfikacji przy użyciu wag. Sygnał wyjściowy neuronu k wykazuje następującą zależność:

$$y = \varphi\left(\sum_{i=1}^{n} w_{ki} x_i\right)$$

gdzie: w_{ki} jest wagą neuronu, φ to funkcja aktywacji, która przetwarza wynik z pierwszego etapu działania sieci, czyli iloczynu skalarnego wektora wag i wektora danych wejściowych (Haykin, 2010) (Stęgowski 2004).

3.1.2. Funkcja aktywacji

Funkcja aktywacji jest wykorzystywana do bardziej efektywnego szkolenia sieci neuronowej i definiowania formatu, który przyjmie sygnał wyjściowy. Konstruując sieć neuronową, w zależności od logiki problemu i oczekiwanego wyniku, dobierana jest odpowiednia funkcja aktywacji. Istnieje wiele typów tych funkcji, z których najpopularniejsze to (Haykin, 2010) (Stęgowski 2004):

- **Funkcja progowa** – tego typu funkcję, nazywaną również modelem McCullocha-Pittsa, opisuje następujące równanie:

$$\varphi = \begin{cases} 1, \text{ dla } v \geq 0 \\ 0, \text{ dla } v < 0 \end{cases}$$

- **Funkcja sigmoidalna** – funkcja, której wykres kształtem przypomina literę „S", jest najczęstszą formą funkcji aktywacji używaną do konstrukcji sieci neuronowych. Jest to ściśle zwiększająca się funkcja, która wskazuje równowagę między liniowym i nieliniowym zachowaniem. Funkcję sigmoidalną możemy przedstawić następująco:

$$\varphi(v) = \frac{1}{1 + \exp(-av)}$$

gdzie α jest parametrem nachylenia funkcji. Przez jego zmianę możemy otrzymać funkcję o różnych kątach nachylenia. Kiedy parametr nachylenia będzie

zbliżał się do nieskończoności, funkcja sigmoidalna stanie się funkcją progową. Zasadniczą różnicą między tymi dwiema funkcjami jest to, że funkcja progowa nie jest zmienna i przyjmuje wartość 1 lub 0, podczas gdy funkcja sigmoidalna wyróżnia się zmiennością oraz wartościami ciągłymi w przedziale od 0 do 1.

• **Funkcja tangens hiperboliczny** – jest funkcją bardzo podobną do funkcji sigmoidalnej, z tą różnicą, że może dać wynik ujemny. Dane wyjściowe mieszczą się w przedziale [–1,1].

$$\varphi(v) = tanh(v)$$

3.1.3. Uczenie sieci neuronowych

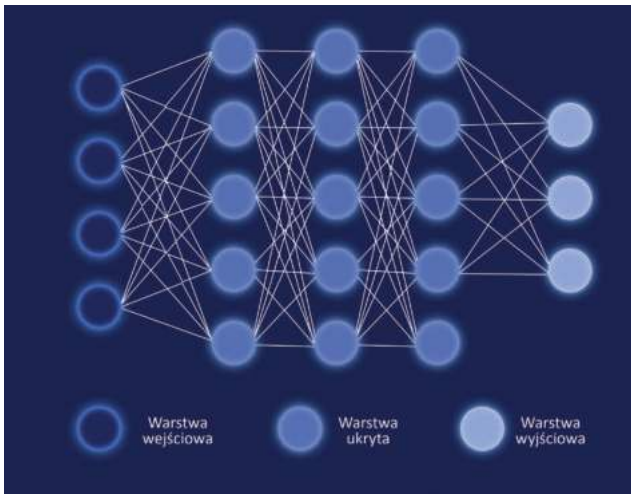
Do uczenia sieci neuronowych stosuje się techniki uczenia nadzorowanego, nienadzorowanego lub uczenia przez wzmacnianie.

Do kontrolowania tego, jak szybko model dopasowuje się do określonego problemu, wykorzystuje się wskaźnik uczenia się (ang. *Learning rate*). Zazwyczaj jest to wartość z przedziału [0,1]. Im mniejsza wartość wskaźnika, tym wolniej sieć się uczy. Należy ostrożnie dobierać wskaźnik uczenia się, ponieważ zbyt szybkie uczenie się sieci, może dać niepoprawne wyniki, natomiast zbyt wolne spowoduje znaczne wydłużenie procesu (Goodfellow, 2016).

Ogólnym celem uczenia sieci neuronowej jest uzyskanie pożądanej odpowiedzi z danych wejściowych. Do realizacji tego celu niezbędnym krokiem jest zmiana wartości wag. Regułą, która pozwala na korektę wag w przypadku uczenia nadzorowanego sieci jednowarstwowych, jest reguła delty (otrzymywana z zależności gradientowej metody najszybszego spadku[3]). W przypadku sieci wielowarstwowych wykorzystuje się algorytm propagacji wstecznej[4], umożliwiający równoczesną zmianę wag we wszystkich warstwach (Stęgowski 2004).

3.2. Głębokie sieci neuronowe

Głębokie sieci neuronowe (ang. *Deep Neural Networks*, DNN), w przeciwieństwie do standardowych sztucznych sieci neuronowych (typowych posiadających do trzech warstw), zawierają wiele warstw ukrytych (Rysunek 15). Tzw. „głęboka" architektura umożliwia skuteczniejsze trenowanie większej liczby parametrów,



Rysunek 15. Deep Neural Network.
Źródło: opracowanie własne.

a tym samym – dokładniej przybliżać funkcje o wysokiej złożoności (Bianchini i Scarselli, 2014).

Istnieje wiele zjawisk, które można skutecznie analizować dzięki głębokim sieciom neuronowym. Przykładem może być obraz, który można przedstawić na różne sposoby, np. jako wektor o wartości intensywności odpowiadającej każdemu pikselowi, jako zbiór krawędzi ciągłych obszarów, obszar o określonym kształcie lub odpowiednio do niego podobny. Zaletą głębokich sieci jest wykorzystanie metody uczenia nienadzorowanego lub częściowo-nadzorowanego i wyodrębnianie funkcji hierarchicznych, dzięki czemu możliwe staje się zastąpienie ich ręcznego wprowadzania (Deng i in., 2014).

Wykorzystując DNN, możliwe stało się rozwiązywanie zadań w takich obszarach, jak:

- **Cyberbezpieczeństwo i bankowość** – wykrywanie różnych zdarzeń, np. wejść nieuprawnionych osób do systemu, czy programów przechwytujących nielegalne dane. Wykrywanie ma na celu poprawę bezpieczeństwa i zwiększenie kontroli nad danymi (Amit i in., 2018) (Mahdavifar i Ghorbani, 2019). Przypadki użycia DNN w cyberbezpieczeństwie i bankowości zostały szerzej opisane w podrozdziale 3.5 Zastosowania sztucznych sieci neuronowych w cyberbezpieczeństwie i bankowości.
- **Zarządzanie relacjami z klientami** – prognozowanie przychodów wynikających z akcji reklamowych (Tkachenko, 2015).
- **Systemy rekomendacji** – przedstawianie ofert o produktach, którymi użytkownik mógłby być zainteresowany z wykorzystaniem zebranych informacji na temat preferencji i cech użytkownika (Elkahky i in. 2015).

[3] Gradientowa metoda najszybszego spadku – algorytm numeryczny wykorzystywany w celu znalezienia minimum funkcji straty, będący modyfikacją metody gradientu prostego.
[4] Algorytm propagacji wstecznej (ang. *Back Propagation*) – metoda uczenia sieci wielowarstwowej, w której błąd ostatniej warstwy jest przesyłany wstecz i wykorzystywany do zmiany wartości wag w poprzednich warstwach.

- **Przetwarzanie mowy** – rozpoznawanie ludzkiej mowy i tworzenie wzorców głosowych (Mohamed i in. 2009) (Deng i in., 2014) (Hannun i in., 2014).

- **Bioinformatyka** – przewidywanie adnotacji ontologicznych genów i zależności funkcji genów (Chicco i in., 2014).

- **Rozpoznawanie obrazów** – przetwarzanie obrazów w celu rozpoznawaniu występujących na nich obiektów i ich klasyfikacja (Ciresan i in. 2012).

- **Przywracanie obrazów** – odzyskanie wyraźnego obrazu z rozmytych, niskiej jakości zdjęć (R. Chen i in. 2019).

- **Usprawnianie wyszukiwania i poprawa skuteczności reklam** – automatyzacja podejmowania decyzji operacyjnych przyspieszających wprowadzanie reklam (Katsov, 2017).

- **Zautomatyzowany czat** (ang. *Chatbot*) – konwersacja z użytkownikiem za pomocą programu komputerowego używającego języka naturalnego, sprawiającego wrażenie rozmowy z żywym człowiekiem (Janarthanam, 2017).

- **Zarządzanie zasobami i zasobami ludzkimi** – wykorzystywane do poprawy wydajności oraz wzrostu produktywności i siły roboczej (Eubanks 2018).

3.3. Konwolucyjne sieci neuronowe

Konwolucyjne sieci neuronowe (ang. *Convolutional Neural Networks*, CNN) wykorzystują dane wejściowe zapisane w tablicach. Można je przedstawić jako wyuczone parametry filtra lub zestawu filtrów, które najlepiej wydobywają informacje ze zbioru danych w ramach zadanego problemu. Konwolucyjne sieci neuronowe bardzo dobrze sprawdzają się w przetwarzaniu obrazów, jednak stosuje się je również do przetwarzania innych struktur danych.

Konwolucyjne sieci neuronowe (Rysunek 17) składają się z trzech odrębnych warstw:

• **konwolucyjnej** (*Convolutional layer*) – głównej warstwy sieci CNN, która jest w stanie stopniowo filtrować różne podzbiory danych uczących i wyostrzać ważne cechy w procesie dyskryminacji, następnie wykorzystywać je w rozpoznawaniu lub klasyfikowaniu wzorców.

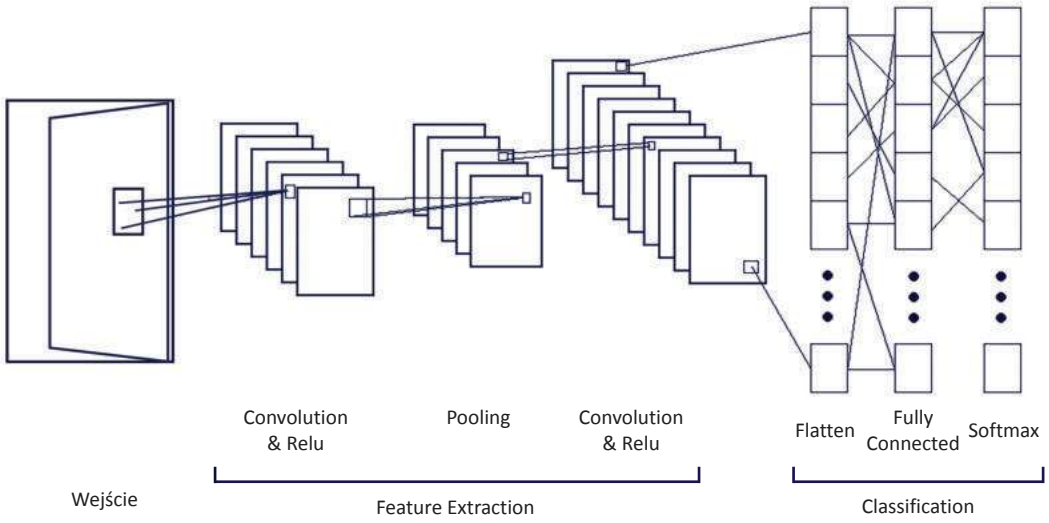
• **łączącej** (*Pooling layer*) – służącej do progresywnego zmniejszania rozmiaru przestrzennego oraz redukowania liczby cech i złożoności obliczeniowej sieci. Umieszczona jest pomiędzy warstwami konwolucyjnymi, najczęściej występuje kilkukrotnie.

• **klasyfikującej** (*Classification layer*) – obliczającej wynik klasyfikacji; składa się z warstw:

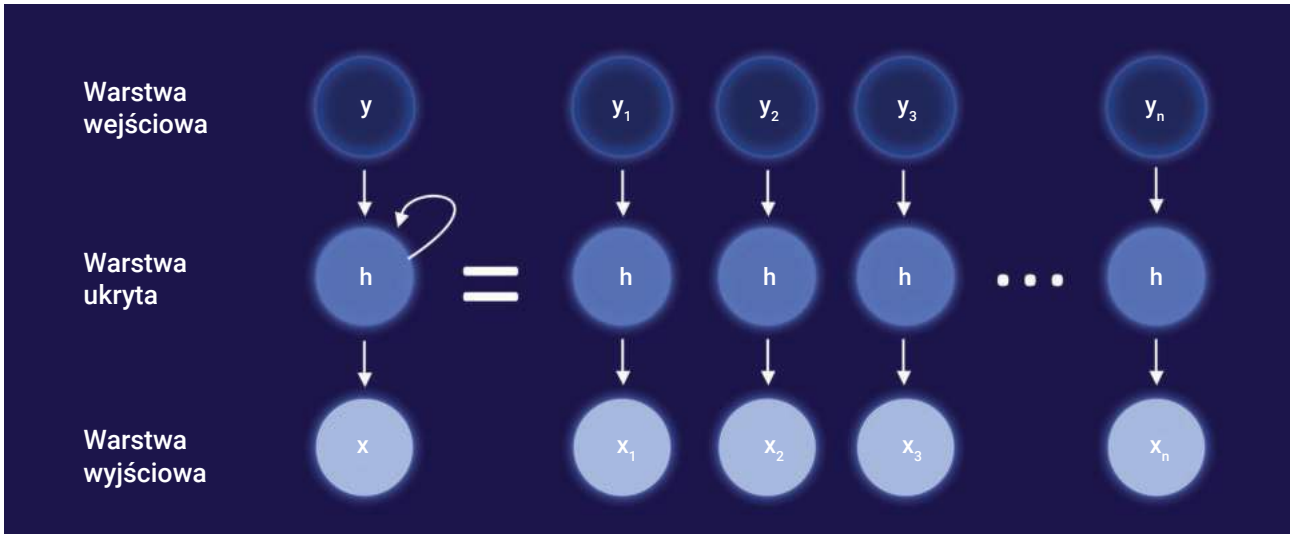
• **spłaszczonej** (*Flatten*) – korzystającej z wyjść wcześniejszych warstw, zmieniając je w pojedynczy wektor;

• **w pełni połączonej** (*Fully Connected*) – pobierającej dane z wcześniejszej warstwy i stosującej wagi, aby przewidzieć prawidłową etykietę;

• **softmax** – konwertującej wynik ostatniej warstwy na rozkład prawdopodobieństwa.



Rysunek 16. Konwolucyjna sieć neuronowa.
Źródło: (Berman i in, 2019, s. 7]



Rysunek 17. Rekurencyjne sieci neuronowe.
Źródło: opracowanie własne.

CNN może być także użyte w przypadku wykrywania szkodliwego oprogramowania. W zaproponowanym przez autorów (Vu I in., 2019) rozwiązaniu, obrazy są przetwarzane przez głęboką konwolucyjną sieć neuronową, klasyfikującą dane wejściowe na łagodne lub złośliwe kategorie. Skuteczność proponowanego przez autorów modelu wynosi 99,14% (w oparciu o dane testowe).

3.4. Rekurencyjna sieć neuronowa

Rekurencyjna sieć neuronowa (ang. *Recurrent Neural Network*, RNN), której budowa została zaprezentowana na Rysunku 17, rozszerza tradycyjne sieci neuronowe o możliwość przyjmowania sekwencji wejściowych o zmiennej długości. Sieci RNN są nazywane rekurencyjnymi, ponieważ wykonują to samo zadanie dla każdego elementu sekwencji, a dane wyjściowe zależą od poprzednich obliczeń. Z tego powodu sieci te świetnie się sprawdzają w rozwiązywaniu skomplikowanych problemów występujących w procesach przetwarzania mowy i języka, a także szeregów czasowych (Berman i in. 2019).

Trenowanie sieci RNN jest zazwyczaj trudniejsze z powodu łatwego zanikania lub eksplozji gradientu[5]. Pierwszy z wymienionych problemów powstaje, gdy dodawane są kolejne warstwy przy użyciu niektórych funkcji sieci neuronowych, wtedy gradienty funkcji straty (ang. *loss function*) zbliżają się do zera, utrudniając tym samym trenowanie sieci. Z kolei zjawiska tzw. eksplozji gradientu powstają, gdy kumulują się duże gradienty błędów, powodujące bardzo duże aktualizacje wag modelu sieci w czasie trenowania. Prowadzi to do niestabilności modelu, uniemożliwiając tym samym uczenie się modelu na podstawie danych treningowych (Berman i in. 2019).

Aby umożliwić RNN rozwiązywanie problemów, które wymagają pamięci długoterminowych, opracowana została LSTM (*Long Short-Term Memory*). Jednostki LSTM zawierają strukturę zwaną komórką pamięci, gromadzącą informacje, które łączą się ze sobą w następnym kroku czasowym. Wartości komórki pamięci są powiększane o nowe dane wejściowe i zawierają tzw. bramkę zapominającą, która waży nowsze lub starsze informacje w zależności od potrzeb. W rezultacie, sieci RNN są obecnie skutecznie stosowane w przewidywaniu następnego słowa w zdaniu, rozpoznawaniu mowy, rozpoznawaniu obrazów i tłumaczeniu.

Dzięki szybkiemu rozwojowi sieci RNN zaczęto także budować algorytmy wykorzystywane w cyberbezpieczeństwie do wykrywania intruzów, złośliwego oprogramowania czy oszustw.

Na podstawie modelu opisanego w (Kim i in. 2016) i przeprowadzonych testów wydajnościowych, udowodniona została możliwość użycia głębokich sieci LSTM dla celów wykrywania włamań do systemu (ang. *Intrusion Detection*). Skuteczność tego rozwiązania została oceniona na poziomie aż 96,93%. Wynik ten został osiągnięty na podstawie modelu trenowanego i testowanego na zbiorze „KDD Cup 1999 dataset".

[5] Gradient – wektor określający wartość i kierunek najszybszego wzrostu wielkości skalarnej.

3.5. Zastosowania sztucznych sieci neuronowych w cyberbezpieczeństwie i bankowości

Tabela 4 powstała na podstawie szerokiego przeglądu literatury. Wyróżnione w niej zostały przykładowe przypadki użycia różnych modeli głębokich sieci neuronowych, skutecznie rozwiązujące różnorodne problemy z zakresy cyberbezpieczeństwa i bankowości. Większość proponowanych rozwiązań charakteryzuje się lepszą skutecznością niż metody uczenia maszynowego lub jednowarstwowe sieci neuronowe.

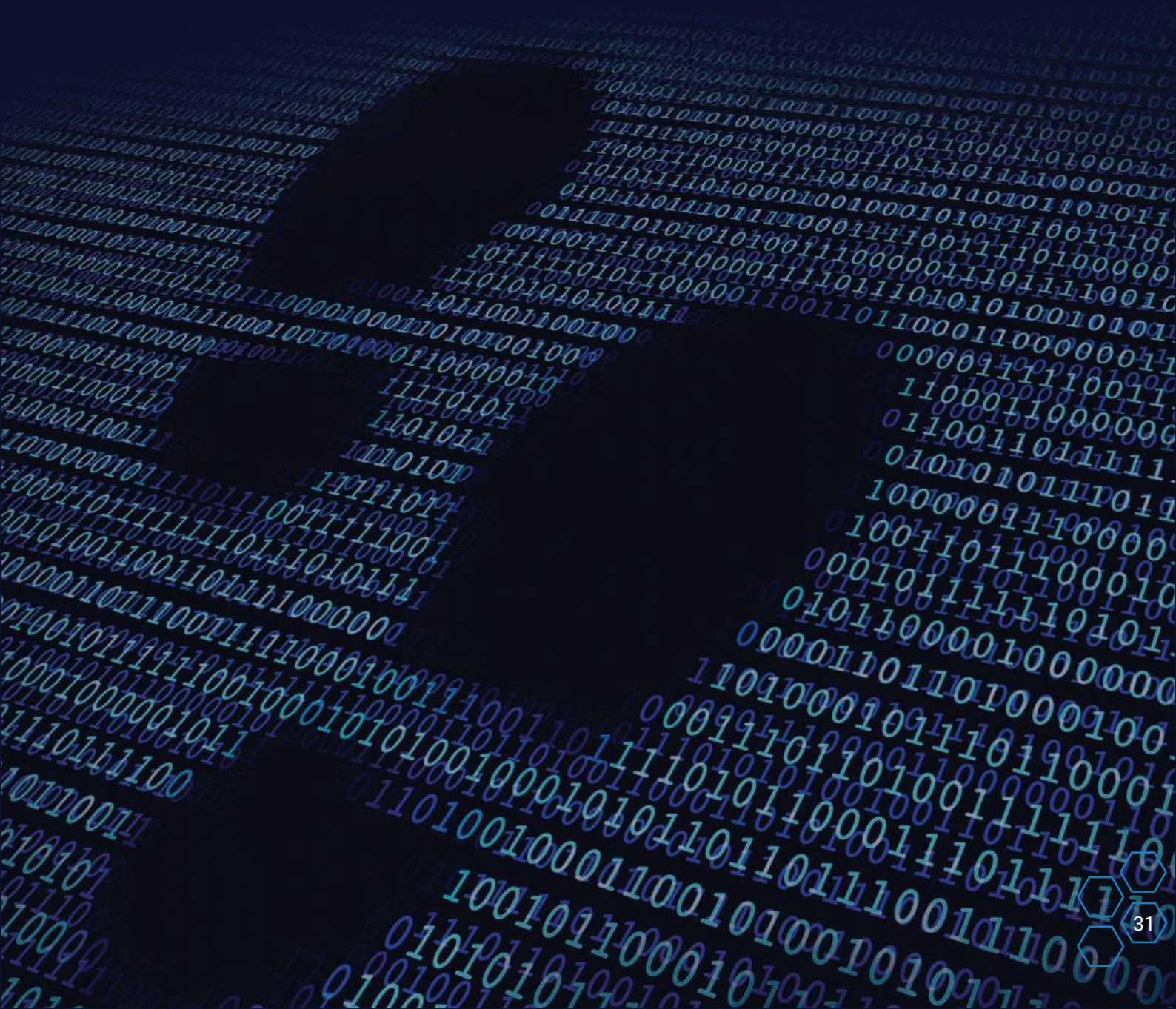
Tabela 4. Zastosowanie sztucznych sieci neuronowych w cyberbezpieczeństwie i bankowości.

Nazwa metody	Zastosowanie	Referencje
RNN	Cyberbezpieczeństwo – wykrywanie złośliwego oprogramowania (malware)	(Babu I in., 2019), (Pascanu i in. 2015)
	Cyberbezpieczeństwo – wykrywanie włamań (intrusion detection)	(Torres i in., 2016), (Babu I in., 2019)
	Cyberbezpieczeństwo – wykrywanie oszustw (fraud detection)	(Babu I in., 2019)]
	Cyberbezpieczeństwo – wyłudzanie informacji (phishing)	(Bahnsen i in. 2017)
	Cyberbezpieczeństwo – wykrywanie zagrożeń wewnętrznych (inside threat detection detection)	(Tuor I in., 2017)
	Bankowość – wykrywanie oszustw (fraud detection)	(Pouramirarsalani i in., 2017)
	Bankowość – zautomatyzowany system udzielania informacji w bankowości	(Oh i in. 2019)
LSTM	Cyberbezpieczeństwo – wykrywanie włamań (intrusion detection)	(Kim i in. 2016)
	Cyberbezpieczeństwo – wykrywanie cyberataków (cyberattacks detection)	(Filonov i in. 2017)
	Cyberbezpieczeństwo – wykrywanie anomalii (anomaly detection)	(Vinayakumar i in., 2017a)
	Cyberbezpieczeństwo – wykrywanie spamu (spam detection)	(Jain I in. 2019)
	Bankowość – wykrywanie oszustw (fraud detection)	(Patel i in. 2019)
	Bankowość – prognozowanie ryzyka bankowego	(Li I in. 2018)
CNN	Cyberbezpieczeństwo – wykrywanie złośliwego oprogramowania (malware)	(Vu I in., 2019)
DNN	Cyberbezpieczeństwo – wyłudzanie informacji (phishing)	(Ra i in., 2018)
	Cyberbezpieczeństwo – detekcja oprogramowania szantażującego (ransomware)	(Vinayakumar I in, 2017b)
	Bankowość – wykrywanie oszustw (fraud detection)	(Heryadi i Warnars, 2017)

Źródło: opracowanie własne.

ROZDZIAŁ IV

Profilowanie behawioralne i ślad cyfrowy



4.1. Profilowanie behawioralne

Każdy ruch użytkownika w sieci wiąże się z pozostawieniem tzw. śladu cyfrowego, gdyż wszelkie akcje wykonywane przez użytkownika generują duże ilości danych. Właściwa analiza tych danych może być źródłem wartości dodanej, np. w postaci informacji na temat użytkowników korzystających z danego portalu bądź systemu.

Profilowanie behawioralne polega na analizie uporządkowanych danych opisujących zachowanie użytkownika, rozpoznawaniu jego cech charakterystycznych i zwyczajów, poznawaniu zainteresowań i intencji, a także opisanie go w interaktywnym środowisku (Foroughi i Luksch, 2018).

Profil behawioralny tworzy się na podstawie działania konkretnego użytkownika, a porównany z innymi profilami, może umożliwić identyfikację podobieństw między nimi i tworzenie profili zbiorczych opisujących grupy użytkowników o wspólnych cechach. Takie podejście jest wykorzystywane w działaniach marketingowych przy analizie grup docelowych i podziale użytkowników na kategorie (segmentacji), a także dostarcza zbiorczych informacji na temat ruchu na stronie. Umożliwia to przede wszystkim głębszą interpretację i poznanie swoich konsumentów, użytkowników sieci.

W literaturze można znaleźć wiele badań na temat tworzenia profili behawioralnych, które opierają się na różnych metodach i algorytmach w zależności od poszukiwanych cech użytkownika (Ortigosa i in., 2014) (Pannell i Ashman, 2010) (Chen i in. 2010) (Su i Zhang, 2006). Metody statystyczne wykorzystują m.in. odchylenie standardowe, średnią ruchomą, sztywno ustawione przedziały dopuszczalnych wartości i tworzą system punktacji (Pannell i Ashman, 2010), który po przekroczeniu danego limitu ostrzega o anomalii. Metody te pozwalają również na przedstawienie sesji użytkowników za pomocą wektorów (Chen i in. 2010) składających się z poszczególnych wartości statystycznych. Wektory te są sposobem na reprezentację sesji i późniejszego porównania lub analizy skupień.

Kolejną grupą metod do tworzenia profili behawioralnych są metody uczenia maszynowego, które pozwalają odkryć jeszcze większą ilość informacji o użytkowniku, ponieważ oprócz statystyk, pozwalają zidentyfikować schematy ruchu. Najczęściej stosowane metody to:

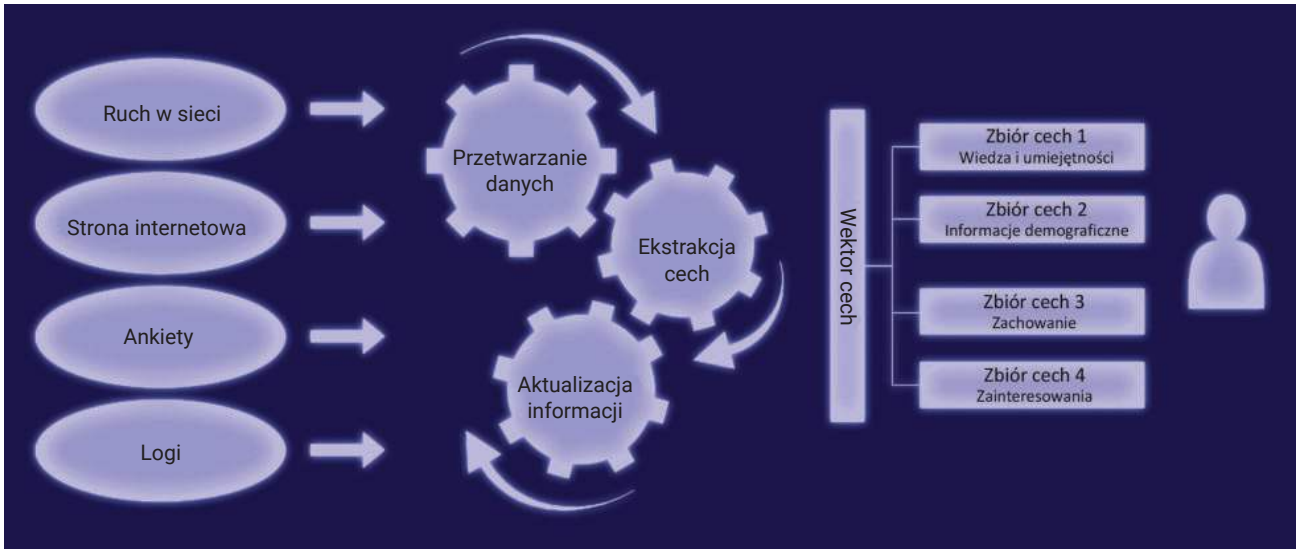
- sieci Bayesowskie (Su i Zhang, 2006): które są modelem prawdopodobieństwa zdarzeń i pozwalają przewidzieć prawdopodobieństwo zdarzenia biorąc pod uwagę poprzednie akcje,
- k-najbliższych sąsiadów (Xiong i in., 2007) (Alazab, 2015) – jest algorytmem uczenia nadzorowanego, który określa daną cechę patrząc na sąsiadujące cechy i przyporządkowuje jej etykietę,
- drzewa decyzyjne (Pachidi i in. 2014), które szacują prawdopodobieństwo przynależności danego elementu do danej klasy,
- reguły asocjacyjne np. algorytm Apriori (Xiang i Gong, 2008), znajdujący zależności pomiędzy wszystkimi akcjami i reguły ich występowania.

Przykładowy, ogólny model tworzenia profilu behawioralnego (Rysunek 18) składa się z trzech głównych kroków:

1. zbieranie surowych danych z platformy, która jest monitorowana,
2. przetwarzanie surowych danych do jednolitego formatu oraz wstępne wnioskowanie podstawowych informacji (np. pora dnia, rodzaj lokalizacji dla odwiedzonej strony, stopień zagrożenia wykonywanych czynności itd.) z wykorzystaniem metod statystycznych lub uczenia maszynowego,
3. sprowadzenie cech do postaci wektora będącego profilem behawioralnym, który można wykorzystać do późniejszej analizy i traktować jako wzór zachowania danego użytkownika.

Teoretycznie im większa ilość informacji zebrana na temat użytkownika, tym lepiej. Jednak w praktyce dobór zbieranych cech opisujących użytkownika zależy od konkretnego zastosowania, możliwości obliczeniowych oraz uwarunkowań prawnych. Jest to szczególnie istotne przy dużej liczbie użytkowników, gdzie ilości generowanych danych mogą przewyższać zdolności obliczeniowe infrastruktury i wydłużać czas reakcji w przypadku wykrywania anomalii.

W cyberbezpieczeństwie monitorowanie ruchu na stronie może pomóc przy autentykacji użytkowników oraz rozpoznawaniu ich intencji. Przy rosnącym udziale botów w ruchu sieciowym (42,2% całego ruchu w sieci generowane jest przez boty (Distil Networks 2019)) konieczna jest ich weryfikacja, możliwość odróżnienia ich od człowieka. Obecne wyzwanie związane z obroną przed atakami cybernetycznymi polega na tym, że szybkość i liczba pojawiających się zagrożeń często



Rysunek 18. Ogólny model tworzenia profilu behawioralnego.
Źródło: opracowanie na podstawie (Chen i in. 2019).

przewyższa zdolności obronne skoncentrowane na działaniu człowieka (Chomiak-Orsa i in. 2019). Dlatego też nowe podejście oparte na tworzeniu profilu behawioralnego metodami sztucznej inteligencji może przyczynić się do zwiększenia poziomu bezpieczeństwa w sieci. Dodatkowo wraz ze wzrostem zagrożeń ze strony metod Social Engineering[6], które wykorzystują czynnik ludzki jako lukę w zabezpieczeniach, profilowanie behawioralne odpowiada na wyzwania związane z monitorowaniem zachowania użytkowników i wykrywaniem anomalii. Metody sztucznej inteligencji przy wykorzystaniu profili behawioralnych potrafią zidentyfikować próbę ataku typu Social Engineering jeszcze przed jego uruchomieniem (Aldawood i Skinner, 2019), dzięki możliwości odwołania się do wcześniejszego (typowego) zachowania użytkownika.

Istnieją dwie podstawowe metody wykrywania niebezpieczeństw w sieci (Pannell i Ashman, 2010). Pierwsza z nich, tzw. *rule-based system*, oparta jest na wcześniej zdefiniowanych manualnie regułach opisujących możliwe zachowania użytkownika. Druga, tzw. *anomaly-based system*, opiera się na wykrywaniu anomalii, gdzie złośliwe zachowania nie są wcześniej zdefiniowane, ale nowe obserwacje porównuje się z wcześniejszymi aktywnościami, a odstępstwa od norm (anomalie) generują alerty ostrzegające. Według naukowców Kanadyjskiego Instytutu Cyberbezpieczeństwa, profilowanie behawioralne i tworzenie modeli użytkowników jest najbardziej użyteczną metodą ich monitorowania i wykrywania anomalii (Chen i in. 2019) (Pannell i Ashman, 2010). Oprócz poznania zainteresowań i specyficznych wzorców dla każdego z użytkowników, możliwe jest również przewidywanie na podstawie tendencji jego zachowania i mierzenie prawdopodobieństwa poszczególnych aktywności, biorąc pod uwagę historię. Umożliwia to również szacowanie potencjalnego niebezpieczeństwa lub ataku ze strony danego użytkownika, co można następnie wykorzystać w celu profilaktycznego przygotowania się na atak. Dostarcza również wielu cennych informacji do podejmowania decyzji związanych z cyberbezpieczeństwem.

Profilowanie behawioralne znajduje swoje zastosowanie również w kryminalistyce. Tzw. profilowanie kryminalne jest procesem określania charakterystyki sprawcy przestępstwa na podstawie dostępnych danych. Profilowanie behawioralne wykorzystywane jest również w ochronie infrastruktury krytycznej, w których może dojść do ataku, np. na lotniskach, gdzie jego głównym celem jest identyfikacja potencjalnych przestępców (Kirschenbaum i in. 95).

W marketingu metody oparte na profilowaniu behawioralnym pozwalają na precyzyjne targetowanie klientów. Umożliwia to personalizowanie wyświetlanej treści w czasie rzeczywistym, nawet dla znacznej liczby konsumentów. Nie byłoby to możliwe bez automatyzacji, gdyż metody oparte na heurystyce[7] wymagają ingerencji człowieka i nie są możliwe do zastosowania na dużą skalę (Talbot, 2019).

[6] Social Engineering – stosowanie środków psychologicznych i metod manipulacji mających na celu wyłudzenie określonych informacji lub nakłonienie ofiary do realizacji określonych działań.

[7] Metody heurystyczne – ogół sposobów i reguł postępowania służących podejmowaniu najwłaściwszych decyzji w skomplikowanych sytuacjach, wymagających analizy dostępnych informacji, a także przewidywania zjawisk przyszłych; oparte na twórczym myśleniu i kombinacjach logicznych.

4.2. Zastosowanie profilowania behawioralnego

Podsumowując, profilowanie behawioralne dostarcza wielu możliwości nadzoru ruchu w sieci (Tabela 5). Jako odpowiedź na zagrożenia płynące z metod Social Engineering, umożliwia ochronę przed wcześniej trudnymi do wykrycia wyłudzeniami. Główną zaletą jest personalizacja każdego użytkownika sieci oraz skalowalność narzędzi tego typu. Ponadto analizy zachowania nie byłyby możliwe do wykonania manualnie, a profilowanie behawioralne umożliwia ich automatyzację na dużą skalę, zależną tylko od wydajności infrastruktury. Dodatkowo według The Economist (The Economist, 2017) oraz Forbes (Bhageshpur, 2015) dane są obecnie nowym najważniejszym surowcem, a profilowanie wykorzystuje ich często niedoceniony potencjał.

Tabela 5. Zastosowanie profilowania behawioralnego w cyberbezpieczeństwie i usługach finansowych.

Nazwa metody	Opis zastosowania	Referencje
Metody statystyczne	Tworzenie profili behawioralnych użytkowników stron bankowych	(Pannell i Ashman, 2010), (Chen i in. 2010)
	Wykrywanie anomalii w zachowaniu użytkowników na stronach lub platformach bankowych	(Pannell i Ashman, 2010), (Chen i in. 2010)
Sieci Bayesowskie	Przewidywanie zachowania użytkowników w celu optymalizacji wykorzystania infrastruktury banku oraz przewidywania zagrożenia	(Su i Zhang, 2006)
K-najbliższych sąsiadów	Klasyfikacja użytkowników ze względu na ich cechy	(Xiong i in., 2007)
	Wykrywanie anomalii w zachowaniu użytkowników	(Alazab, 2015)
Drzewa decyzyjne	Analiza przyczyn zachowania użytkowników oraz jego przewidywanie	(Pachidi i in. 2014)
Reguły asocjacyjne	Określanie typowych zachowań użytkowników	(Xiang i Gong, 2008)

Źródło: opracowanie własne.

4.3. Ślad cyfrowy

Podczas każdorazowej sesji na witrynie internetowej użytkownik pozostawia bardzo dużo cyfrowych informacji, które po zebraniu w całość, nazywane są śladami lub odciskami cyfrowymi. Istnieją też pliki cookie, które za przyzwoleniem użytkownika, gromadzą w sposób jawny dane o użytkowniku i sesji w przeglądarce.

Ślad cyfrowy to odcisk palca maszyny lub przeglądarki. Jest to informacja zebrana o zdalnym urządzeniu komputerowym w celu jego identyfikacji. Opiera się na specyficznej i unikatowej konfiguracji, z której w większości przypadków jest tworzony identyfikator. Ślady cyfrowe mogą być użyte do pełnego lub częściowego rozpoznawania indywidualnych użytkowników. Techniki pobierania śladów cyfrowych dzielą się na aktywne i pasywne. Podział zależy od tego, w jaki sposób pożądane dane są uzyskiwane.

Aktywne metody polegają na komunikacji z urządzeniem użytkownika wysyłając sekwencje sondujących/pomiarowych pakietów na np. jego komputer, które mają na celu wymusić odpowiedź ze strony jego maszyny. Następnie informacje o aktywności zostają odesłane do nadawcy, czyli podmiotu zbierającego odciski cyfrowe. Później rozpoczyna się ich analiza oraz porządkowanie. Tego typu podejście jest jednak łatwo wykrywalnym sposobem do gromadzenia danych o użytkownikach sieci.

Metody pasywne korzystają z danych wysyłanych przez użytkownika, które nie są wygenerowane

poprzez interakcję z pakietami pomiarowymi. Opierają się na monitorowaniu, gromadzeniu i analizowaniu danych przekazywanych przez użytkownika. Te metody wykorzystują przede wszystkich założenia *sniffingu* (tj. polegającego na odczytywaniu danych np. przez program komputerowy, nieprzeznaczone dla tego strumienia danych), przez co ich detekcja jest bardzo trudna. Użytkownik za każdym razem, wysyłając zapytania i pakiety do strony, udziela informacji o swoim urządzeniu i konfiguracji, które mogą być bez jego wiedzy zbierane i gromadzone w bazie danych właściciela strony.

Serwisy lub programy do tworzenia cyfrowych odcisków w znacznej części wykorzystują pasywne metody ze względu na ich bierność i trudną detekcję. Tworzone identyfikatory są uzyskiwane za pomocą kombinacji danych z poniższego zbioru:

- Adres IP
- User Agent String[8]
- Strefa czasowa klienta
- Nagłówki żądań HTTP
- Informacje o urządzeniu klienta:
  - Rozdzielczość ekranu
  - Głębia koloru
  - System operacyjny
  - Wsparcie dotykowe
  - Język
- Dane Flash zapewnione przez wtyczkę Flash
- Lista zainstalowanych czcionek (flash, javascript)
- Lista zainstalowanych wtyczek
- Lista typów MIME[9]
- Aktualny czas, w którym zostały pobrane dane
- Przyzwolenie na pliki cookie
- Wykorzystanie pamięci lokalnej
- Wykorzystanie pamięci sesji
- Lista poprzednio odwiedzanych stron internetowych.

Istniejące metody do analizy śladu cyfrowego umożliwiają identyfikację użytkownika dysponując wyłącznie podzbiorem danych wymienionych wyżej. Techniki te są często stosowane w praktyce. Ślad cyfrowy jest oparty na danych z przeglądarki, systemu operacyjnego i zainstalowanym sprzęcie graficznym. Techniki śledzenia aktywności użytkowników w przeglądarkach internetowych bardzo szybko się rozwijają i osiągnęły wysoki poziom skuteczności. Równocześnie, nie istnieją narzędzia, które w pełni ukrywałyby ślad cyfrowy użytkownika.

[8] Przechowuje informację o używanej przeglądarce, jej wersji oraz systemie operacyjnym. Przykładowy UAS: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US) AppleWebKit/525.19 (KHTML, like Gecko) Chrome/1.0.154.53 Safari/525.19

[9] Internet media type (ang. *Multipurpose Internet Mail Extensions*), zwany także typem MIME, jest to dwuczęściowy identyfikator formatu plików w Internecie.

Jednym ze sposobów na rozszerzenie zakresu dodatkowych danych do budowy śladu cyfrowego są metody wykorzystujące renderowanie obrazu za pomocą elementu *canvas* języka HTM[10] lub WebGL[11] (Javascript API)[12].

Canvas Fingerprinting to technika śledzenia użytkownika za pomocą przeglądarki internetowej, pozwalająca właścicielom witryn internetowych na skuteczne identyfikowanie i badanie klienta, używając w tym celu elementu *canvas* z języka HTML5, zamiast internetowych plików cookies[13]. Canvas Fingerprinting polega na wykorzystaniu luki w elemencie *canvas*. W momencie wejścia użytkownika na stronę, jego przeglądarce jest polecone stworzenie ukrytej linii tekstu lub grafiki 3D, która potem jest renderowana w cyfrowy znak oraz unikatowy tekstowy identyfikator (patrz Rysunek 19).

Rysunek 19. Odcisk cyfrowy uzyskany za pomocą Canvas Fingerprinting.
Źródło: https://medium.com/slido-dev-blog/we-collected-500-000-browser-fingerprints-here-is-what-we-found-82c319464dc9

4.3.1. Zastosowania

Gromadzenie nawet niedużej ilości danych o użytkowniku umożliwia szereg zastosowań opisanych w tej sekcji. Skutki tych działań są zależne od motywów podmiotów zbierających dane.

Identyfikacja użytkownika

W projekcie Panoptlick (Eckersley, 2010) autorzy badali unikatowość przeglądarki. W celu zebrania zestawu danych stworzyli stronę (https://panopticlick.eff.org/), na której odwiedzający pozostawiali swoje dane. Korzystali z następującego zbioru danych:

[10] HTML (hipertekstowy język znaczników) – ustandaryzowany system do oznaczania plików tekstowych w celu nadania czcionki, koloru, grafiki czy hiperłącza do efektów na stronie internetowej.

[11] WebGL (Web Graphics Library) to API języka JavaScript służące do renderowania (rysowania) interaktywnej grafiki 3D i 2D poprzez kompatybilną przeglądarkę bez używania wtyczek.

[12] API – Application Programming Interface (ang. Interfejs Programowania Aplikacji) – umożliwia komunikację między aplikacjami (np. programu z systemem operacyjnym).

[13] Cookie – niewielki plik zapisywany na komputerze użytkownika, w którym przechowywane są ustawienia i inne informacje używane na odwiedzanych przez niego stronach.

- User Agent String
- nagłówek HTTP Accept[14]
- zgoda na pliki cookie
- rozdzielczość ekranu
- strefa czasowa
- lista wtyczek i typów MIME
- czcionki systemowe
- wykorzystanie pamięci lokalnej
- wykorzystanie pamięci sesji.

W czasie badania udało się zgromadzić 470 161 instancji śladów cyfrowych. Analiza wykazała, że podział odcisków cyfrowych zawiera przynajmniej 18.1 bitów entropii, co oznacza, że gdy losowo wybierzemy jedną instancję, w najlepszym przypadku możemy się spodziewać, że w zbiorze 286 777 instancji, tylko jedna będzie dokładnie taka sama. Skuteczność identyfikacji użytkownika wyniosła 83,6%, a 5,3% odcisków było identycznych z innym odciskiem. W przypadku przeglądarek, w których możliwe było użycie Adobe Flash Player[15] lub Java Virtual Machine[16], zanotowano skuteczność na poziomie 94,2%, i tylko 4,8% odcisków mających identyczną strukturę z drugim odciskiem. Badacze w swoich próbkach brali pod uwagę także dynamikę odcisków, gdyż niektóre ustawienia urządzenia/przeglądarki zmieniają się wraz z upływem czasu, jednak używając prostych heurystyk byli w stanie przewidzieć do kogo należy zmieniony ślad cyfrowy.

Śledzenie ruchu klienta w sieci

Serwisy internetowe, w celu zapewnienia bezpieczeństwa, dostarczenia personalizowanych treści, w tym przede wszystkim ukierunkowanej reklamy, również używają technik śledzących użytkownika. Klient korzystając z Internetu, w każdym miejscu zostawia swój ślad cyfrowy, jednakże nie jest on zawsze taki sam. Mnogość aktualizacji systemów operacyjnych lub oprogramowania, a także ingerencja samego użytkownika w konfigurację, jest w stanie modyfikować odcisk cyfrowy nawet co kilka godzin. Przykładem może być metoda FP-STALKER (Vastel i in., 2018), która skupia się na dynamice tych odcisków. W celu powiązania zmodyfikowanych odcisków z tą samą instancją przeglądarki, FP-STALKER używa wariantu opartego na regułach, który jest szybszy, oraz wariantu hybrydowego, który wykorzystuje uczenie maszynowe w celu podwyższenia skuteczności. W projekcie zgromadzono 98 598 śladów cyfrowych przeglądarki pochodzących z 1 905 instancji przeglądarek. Przeprowadzone testy wykazały, że metoda FP-STALKER jest w stanie poprawnie linkować odciski cyfrowe średnio przez więcej niż 51 dni.

Analiza kredytowa

W celu uzyskania kredytu bank ocenia zdolność kredytową klienta m.in. na podstawie wysokości wynagrodzenia, rodzaju umowy o pracę, czasu trwania umowy, ilości posiadanych zobowiązań kredytowych i sposobu ich spłacania. Federal Deposit Insurance Corporation (Berg i in. 2018) poddało obserwacjom dane o zdolności kredytowej z prywatnych agencji ratingowych wraz ze śladem cyfrowym potencjalnych kredytobiorców, na który składały się dane o:

- typie urządzenia,
- kanale, przez który użytkownik dostał się na stronę (np. wyszukiwarka, strona do porównywania cen sprzętu),
- systemie operacyjnym,
- czasie zakupów,
- dostawcy usług email (np. Gmail, Hotmail itd.),
- dwie informacje o emailu wybrane przez użytkownika (zawierające imię i/lub nazwisko oraz czy w emailu zawiera się liczba).

Badania pokazały 10-procentową korelację między danymi z prywatnych agencji ratingowych a predykcją zdolności kredytowej bazującą na odcisku cyfrowym. Dalsza analiza pokazuje, że model predykcyjny wykorzystujący obydwie składowe znacznie przewyższa modele opierające się tylko na jednej z nich. Oprócz tego zaobserwowano wiele dodatkowych reguł, np. odciski cyfrowe, w których system operacyjny to iOS, wskazują na posiadanie przez użytkownika produktu Apple, co z kolei zwiększa szansę, że użytkownik zarabia powyżej średniej, a więc jego zdolność kredytowa i płynność finansowa jest lepsza.

Opisane zastosowania, w pewien sposób naruszają prywatność, ale płyną z nich również korzyści dla klientów/użytkowników, np. lepsza identyfikacja podczas procesu logowania się na stronach lub spersonalizowane oferty reklamowe czy też biznesowe. W oparciu o profilowanie behawioralne implementuje się również systemy autoryzacji, które chronią użytkownika przed próbami przejęcia ich tożsamości czy kradzieżą kont. Przykładem mogą być alerty wysyłane przez system w chwili, gdy logowanie na konto odbywa się z nietypowej lokalizacji, lub konieczność dodatkowej autoryzacji (np. kodem sms), kiedy użytkownik loguje się z „niezaufanego" urządzenia.

Kwestie prawne związane z profilowaniem zostały uregulowanie w RODO (Rozporządzenie

[14] Accept (nagłówek żądania) – informuje, jakie formaty danych akceptuje klient.

[15] Adobe Flash Player – multimedialna wtyczka do przeglądarek internetowych i jednocześnie odtwarzacz animacji rozwijany i dystrybuowany przez firmę Adobe Inc.

[16] Java Virtual Machine – maszyna wirtualna oraz środowisko zdolne do wykonywania kodu bajtowego Javy.

Parlamentu Europejskiego i Rady (UE) 2016/679 z dnia 27 kwietnia 2016 r. w sprawie ochrony osób fizycznych w związku z przetwarzaniem danych osobowych i w sprawie swobodnego przepływu takich danych oraz uchylenia dyrektywy 95/46/WE; Ustawa z dnia 10 maja 2018 r. o ochronie danych osobowych, Dz.U. 2018 poz. 1000). Zgodnie z obowiązującymi przepisami, profilowanie użytkownika jest możliwe wyłącznie po poinformowaniu go o zakresie i celach tego procesu oraz uzyskaniu jednoznacznej, dobrowolnej, konkretnej, opartej na rzetelnej informacji oraz wyraźnie wyodrębnionej zgody. Takie rozwiązania zostały przyjęte, aby chronić prywatność użytkowników oraz ograniczyć np. możliwość wykorzystania profilowania do dyskryminacji cenowej, wykluczenia klientów uznanych za mało wartościowych. Co istotne, RODO dopuszcza przetwarzanie danych zagregowanych i zanonimizowanych bez zgody użytkowników – takie dane nie mogą umożliwiać identyfikacji poszczególnych jednostek (użytkowników), ale mogą dostarczyć wartościowych informacji statystycznych o określonych społecznościach czy grupach użytkowników.

Najważniejszym problemem etycznym, związanym z profilowaniem behawioralnym, jest swoista asymetria informacji. Użytkownicy często nie są świadomi, jakie dane i w jakim celu są gromadzone przez właścicieli witryn. Co więcej, integracja danych z różnych źródeł pozwala na zaawansowane wnioskowanie o cechach czy preferencjach użytkownika, co jest następnie wykorzystywane np. do personalizowania treści reklamowych. W sytuacji, gdy użytkownik nie jest świadomy istnienia tego typu mechanizmów, jest on bardziej podatny na manipulacje. Przykładowo, niektórzy dostawcy treści reklamowych pozwalają na kierowanie ich do użytkowników będących w określonym stanie emocjonalnym (np. „zestresowany", „niepewny swojej wartości") lub stosowanie tzw. d*ark posts* (treści widocznych tylko dla określonych użytkowników) w celu zmotywowania wybranych grup do udziału w głosowaniu (Szymielewicz, 2017).

ROZDZIAŁ V

Wykrywanie anomalii

Anomalia definiowana jest jako dana (punkt, obiekt, wartość w zbiorze), która znacząco odbiega od innych członków próby, w której występuje. W porównaniu do normalnych instancji, występuje ona stosunkowo rzadko w zbiorze danych.

Anomalie można podzielić na (Rysunek 20) (Han, 2011):

• **Punktowe** (ang. *Point anomalies*) – występują, jeżeli pojedynczy obiekt znajduje się daleko od reszty rozkładu. Przykład: Wykrycie oszustwa finansowego na podstawie historii wydatków z konta.

• **Kontekstowe** (ang. *Contextual anomalies*) – występują, jeżeli obiekt jest anomalny tylko w określonym kontekście. Ten typ jest powszechny w szeregach czasowych. Przykład: Próba zalogowania się pracownika do systemu korporacyjnego poza jego godzinami pracy.

• **Zbiorowe** (ang. *Collective anomalies*) – występują, jeżeli grupa obiektów wykazuje nieprawidłowe zachowanie w porównaniu do innych grup obiektów. Indywidualna instancja w grupie anomalnej niekoniecznie sama w sobie jest anomalna. Przykład: Nagły spadek sprzedaży e-commerce na kilka godzin.

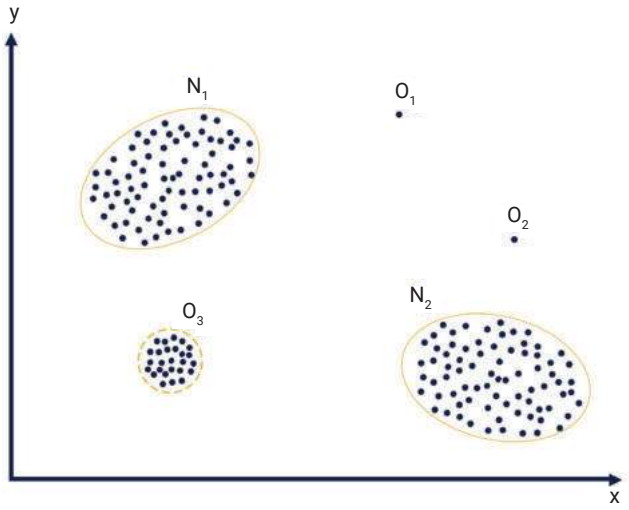
Wykrywanie anomalii (zwanymi też wartościami odstającymi) to technika stosowana do identyfikowania nietypowych wzorców, które nie są zgodne z oczekiwanymi zachowaniami.

Przyczyny występowania anomalii w zbiorach danych są różne. Mogą one być spowodowane m.in.: awariami systemów, włamaniami, oszustwami związanymi z kartami kredytowymi i innymi złośliwymi działaniami (Goldstein i Uchida, 2016).

Wyzwania wiążące się z wykrywaniem anomalii to przede wszystkim (Han, 2011):

• odpowiednie zdefiniowanie dokładnych granic między normalnymi danymi a anomaliami, zwłaszcza że dokładne pojęcie anomalii różni się w zależności od domeny, np. w dziedzinie medycyny małe odchylenie od normy (np. wahania temperatury ciała) może być anomalią, natomiast podobne odchylenie w dziedzinie rynku akcji (np. wahania wartości akcji) można uznać za normalne;

• dynamicznie zmieniający się charakter środowisk monitorowania – występuje to np. w sektorze cyberbezpieczeństwa, w którym to ilość oraz różnorodność ataków na systemy informatyczne stale rośnie;



Rysunek 20. Prezentacja wartości odstających w dwuwymiarowym zestawie danych. Przedstawione zostały dwa normalne zbiory N1 i N2. Pojedyncze punkty znajdujące się daleko od tych zbiorów – O1 i O2 – są anomaliami punktowymi. Grupa punktów O3 to przykład anomalii zbiorowej.
Źródło: (Parmar i Patel, 2017)

• dostosowywanie zachowania osób atakujących system do normalnego zachowania, np. osoba mająca nielegalny dostęp do nieswojego konta bankowego może wypłacać małe kwoty pieniężne, by nie wzbudzać podejrzeń.

5.1. Metody wykrywania anomalii

Metody wykrywania anomalii są zróżnicowane. Można wśród nich wyróżnić przede wszystkim: metody statystyczne, uczenia maszynowego i oparte na grafach. Zostały one omówione w kolejnych podrozdziałach. Ponadto, rozdz. 5.1.4 Wykrywanie anomalii w strumieniach danych – przedstawia specyfikę wykrywania anomalii w strumieniach danych. Wymienione niżej metody są uniwersalne i mogą być zastosowane zarówno w sektorze bankowości, jak i w cyberbezpieczeństwie.

5.1.1. Metody statystyczne

Główną ideą tego podejścia jest pomiar wyniku instancji, który określa odchylenie obiektu od wspólnych właściwości statystycznych rozkładu, w tym średniej, mediany, dominanty i kwantyli. Następnie wynik jest porównywany ze wstępnie określoną wartością progową w celu ustalenia, czy dana instancja jest anomalna. Przykładem takiej metody jest filtr dolnoprzepustowy (ang. *Low pass filter*), który wygładza rozrzut

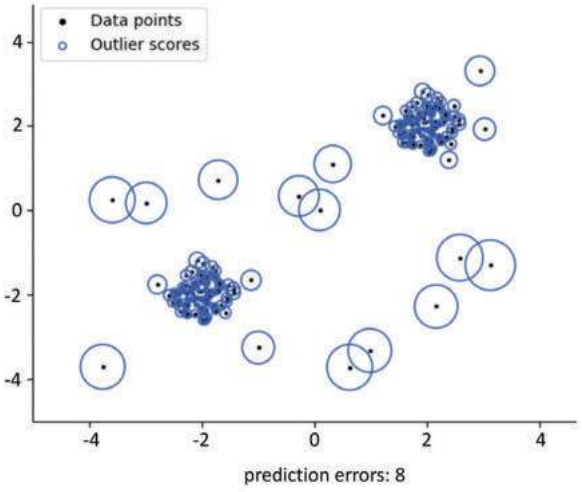
w pomiarach ruchu sieciowego. Jest on wykorzystywany do redukcji fałszywych alarmów wywoływanych wykryciem anomalii (Li i Manikopoulos, 2004).

5.1.2. Metody uczenia maszynowego

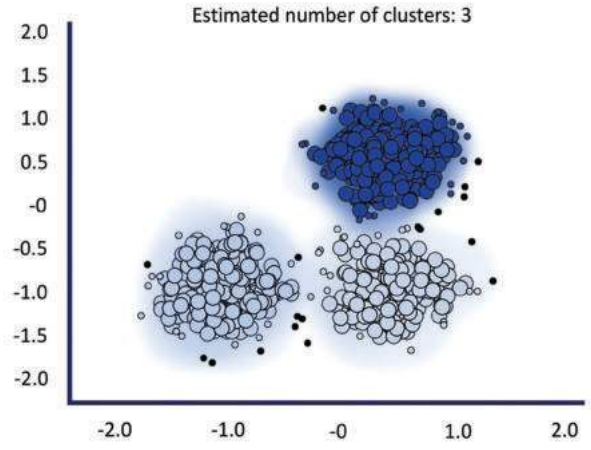
Algorytmy uczenia maszynowego poprawiają swą zdolność odróżniania normalnego zachowania od anomalnego wraz z doświadczeniem, dlatego też są często wykorzystywane przy detekcji obiektów odstających. Wśród nich można wyróżnić:

- Metody oparte na klasyfikacji (opisane w rozdziale 2):
  - SVM – Metoda polegająca na znalezieniu hiperpłaszczyzny, która podzieli zestaw danych i tym samym umożliwi odróżnienie obiektów odstających od normalnych. Szczególnym przypadkiem tej metody jest jednoklasowy (*one-class*) SVM. Algorytm, biorąc pod uwagę nowy obiekt, sprawdza czy mieści się on w granicach normy. Jeżeli obiekt znajduje się poza tą granicą, zostaje uznany za wartość odstającą (Senf, 2006).
  - K-najbliższych sąsiadów (patrz 2.2.1 Metoda k-najbliższych sąsiadów)

- Metody oparte na grupowaniu względem gęstości (ang. *Density-based clustering*). Ich podstawowym założeniem jest to, że gęstość wokół obiektu nieodstającego jest podobna do gęstości wokół jego sąsiadów, natomiast gęstość wokół obiektu odstającego znacznie się różni od gęstości wokół jego sąsiadów. Przykładami algorytmów gęstościowych są m.in.:
  - LOF (*Local Outlier Factor*) – algorytm, który poprzez porównanie lokalnej gęstości obiektu z lokalną gęstością jego sąsiadów identyfikuje regiony oraz punkty, które mają znacznie niższą gęstość niż sąsiedzi. Te punkty są uważane za wartości odstające (Rysunek 21) (Han, 2011)
  - DBSCAN (*Density-Based Spatial Clustering with Noise*) – algorytm szuka obszarów o dużej gęstości i przypisuje do nich klastry, podczas gdy punkty w mniej gęstych regionach są oznaczone jako anomalie (Rysunek 22). W przeciwieństwie do algorytmu k-średnich (opisanego w 2.2.1 Algorytmy analizy skupień), DBSCAN nie wymaga ustalenia liczby klastrów a priori (Hodge i Austin, 2004).



Rysunek 21. Prezentacja wyników algorytmu LOF: dla każdego punktu mierzone jest jego lokalne odchylenie gęstości w odniesieniu do jego sąsiadów. Porównując te gęstości, można zidentyfikować próbki o gęstości znacznie niższej niż ich sąsiedzi, które są uważane za odstające.
Źródło: https://scikit-learn.org/stable/_images/sphx_glr_plot_lof_outlier_detection_001.png



Rysunek 22. Prezentacja wyników algorytmu DBSCAN: oznaczone są trzy obszary o dużej gęstości – klastry. Pojedyncze punkty są interpretowane jako anomalie.
Źródło: https://scikit-learn.org/stable/_images/sphx_glr_plot_dbscan_001.png

5.1.3. Metody oparte na grafach

Grafy w naturalny sposób reprezentują współzależności poprzez wprowadzenie połączeń między powiązanymi obiektami. Obiektami odstającymi w grafach mogą być węzły, połączenia, jak i konkretne fragmenty grafu, które wykazują nietypowe właściwości. Grafy możemy podzielić na (Akoglu, 2015):

- Statyczne
  - Grafy proste (ang. *Plain graphs*) składają się tylko z węzłów i połączeń między nimi. W przypadku grafów prostych jedyną posiadaną informacją jest struktura grafu. Metody służące do wykrywania obiektów odstających można podzielić na dwie kategorie: metody oparte na strukturze (ang. *Structure-based methods*) oraz na społeczności (ang. *Community-based methods*). Pierwsze z nich wykorzystują reprezentację grafu do wyodrębnienia cech strukturalnych, które są używane wraz z innymi cechami wydobytymi z dodatkowych źródeł danych w celu wykrycia wartości odstających. Ideą drugiego podejścia jest znalezienie gęsto połączonych grup „sąsiednich" węzłów lub krawędzi mających połączenia między społecznościami. Za anomalię są uznawane węzły lub krawędzie, które nie należą bezpośrednio do jednej konkretnej społeczności. Zarówno metody oparte na strukturze, jak i na społeczności znajdują zastosowanie przy wykrywaniu spamu i złośliwego oprogramowania (ang. *malware*) w sieci. Techniki te zakładają, że link między dwoma stronami oznacza zaufanie między nimi. Ponadto, jeżeli strona docelowa jest znana jako strona ze spamem lub zawierająca niebezpieczne treści, wówczas uznają daną stronę za potencjalnie niebezpieczną.

  - Grafy atrybutowe (ang. *Attributed Graphs*) posiadają węzły i/lub połączenia opisane atrybutami określającymi ich cechy. Przykładem mogą być sieci społecznościowe. Załóżmy, że istnieje sieć, w której grupa ludzi, która ma podobny gust jeśli chodzi o filmy, podczas gdy większość grup społecznych w tej sieci ma ten gust bardzo zróżnicowany. Taka grupa jest uważana za anomalię. Metody detekcji anomalii w tego typu grafach są takie same, jak w przypadku grafów prostych. Metody oparte na strukturze szukają tych części grafu, które są nietypowe, np. pod względem łączeń, atrybutów. Celem drugiego podejścia jest identyfikacja tych węzłów w grafie, których wartości atrybutów znacznie odbiegają od innych członków społeczności do których należą. Na przykład palacz w społeczności zawodników piłki nożnej, którzy nie są palaczami, jest przykładem obiektu odstającego (Akoglu, 2015). Grafy atrybutowe są wykorzystywane bardzo często w marketingu docelowym (ang. *target marketing*) oraz systemach rekomendacji (Li, 2013), dlatego też, mogą one okazać się bardzo przydatnym narzędziem w bankowości umożliwiającym łatwiejsze dotarcie do klientów.

- **Dynamiczne**

Grafem dynamicznym jest graf, w którym zachodzą zmiany. Mogą one polegać na wstawianiu lub usuwaniu krawędzi, wierzchołków czy też zmianie wartości z nimi związanych, np. wag krawędzi. Problem wykrywania anomalii w przypadku grafów dynamicznych polega na znalezieniu sygnatury czasowej (ang. *timestamp*) odpowiadającej zmianie lub zdarzeniu oraz węzłów, krawędzi lub części grafów, które najbardziej przyczyniają się do zmiany (Henzinger, 2018). Grafy dynamiczne znajdują zastosowanie w wykrywaniu oszustw finansowych poprzez znajdowanie podejrzanych transakcji.

5.1.4. Wykrywanie anomalii w strumieniach danych

Strumień danych to zbiór obserwacji zarejestrowanych w odstępach czasu. Zazwyczaj są to dane zapisywane w jednakowym odstępie czasu, jednak nie zawsze musi tak być. Przykładem są sieci czujnikowe, w których dane generowane są w zależności od zmian zachodzących w środowisku zewnętrznym. Są one stosowane m.in. w monitorowaniu stanu zdrowia pacjentów, inteligentnych budynkach, gdzie dane generowane są przez czujniki IoT (Internet of Things) oraz w bankowości (np. dane z giełdy, które dostarczane są na bieżąco). Wspólną cechą dla wszystkich podejść strumieniowego przesyłania danych jest to, że przetwarzają one dane w momencie ich otrzymania, dlatego też, im więcej obiektów dociera w krótszych okresach, tym szybciej musi zostać podjęta decyzja o wykryciu wartości odstających (Assent i in, 2012).

Przykładem metody statystycznej służącej do detekcji anomalii w strumieniach danych jest metoda DTW (ang. *Dynamic Time Warping*) (Duraj i Chomątek, 2019) (Naval i in. 2014). Algorytm ten jest wykorzystywany do mierzenia podobieństwa między dwoma seriami (strumieniami danych), które mogą być różnej długości, a następnie wykrywa wartości odstające po określeniu dopuszczalnego zakresu wartości (Diab i in. 2019).

Inną metodą umożliwiającą wykrywanie anomalii w strumieniach danych jest tzw. HTM (ang. *Hierarchical Temporal Memory*). Jedną z zasadniczych zalet jest to, że metody oparte na HTM nie wymagają danych uczących ani osobnego etapu uczenia. Automatyczne budowanie modelu oraz jego trenowanie eliminuje potrzebę ręcznego definiowania i utrzymywania modeli oraz zestawów danych, a to znacznie skraca czas użytkownika (Ahmad, Purdy, 2016). Algorytm HTM działa w sposób następujący: dla każdego nowo otrzymanego obiektu oblicza wynik anomalii. Jeżeli został on poprawnie przewidziany, to wynik anomalii jest równy zero, jeżeli nie, to wynosi on jeden. Obiekty, które zostały przewidziane częściowo, mają wartość z przedziału (0, 1). Im większa wartość, tym obiekt jest bardziej uznawany za odstający. Na początku

trenowania modelu większość obiektów będzie miała wysoką wartość, ponieważ będą one nowe. Wraz z uczeniem się algorytmu, wynik anomalii będzie się zmniejszał, dopóki nie nastąpi zmiana w strumieniu wzorców.

Kolejną metodą jest algorytm RCF (ang. *Random Cut Forest*), który został wprowadzony przez firmę Amazon. Algorytm ten, dla każdego punktu danych, przypisuje ocenę anomalii (ang. *anomaly score*). Niskie wartości oceny wskazują, że punkt danych jest uważany za „normalny". Wysokie wartości wskazują na obecność anomalii w danych (Guha i in., 2016).

5.2. Zastosowanie detekcji anomalii

Anomalia to obiekt uznawany za niepasujący do innych członków grupy w której występuje. Zarówno firmy, jak i klienci mogą ponieść duże straty na skutek niewykrycia anormalnych zdarzeń. Codziennie występuje wiele sytuacji, w których ludzkie życie i majątek są narażone. Przykładem poważnych konsekwencji może być niewykrycie choroby u pacjenta lub niezauważenie zaburzeń linii produkcyjnych w fabryce samochodów. Wykrywanie anomalii jest wykorzystywane na wielu płaszczyznach i w wielu branżach (Alla i Adari, 2019).

Przykładem jest sektor bankowości. Operacje bankowe obejmują wiele codziennych, okresowych i nieokresowych transakcji dokonywanych przez lub mających wpływ na wielu interesariuszy, takich jak pracownicy, klienci, dłużnicy i podmioty zewnętrzne. Złożony charakter tych działań i transakcji wymaga ciągłego monitorowania, aby zapewnić, że ani bank, ani jego interesariusze nie zostaną dotknięci zdarzeniami, które mogłyby im zaszkodzić. W tym przypadku detekcja anomalii jest niezwykle istotna, ponieważ wczesne wykrycie anormalnych zdarzeń może znacznie złagodzić potencjalnie negatywne ich skutki, a w niektórych przypadkach aktywnie temu zapobiec. Metody wykrywania anomalii można wykorzystać w (Mohan i Mehrotra, 2017):

- **Śledzeniu zachowań**

Śledzenie zachowań klientów może dostarczyć wskazówek pomocnych w identyfikowaniu anormalnych zachowań. Przykładem mogą być sytuacje, w których wydatki klientów zaczynają znacznie różnić się od wcześniejszej historii transakcji, co może być konsekwencją kradzieży tożsamości klienta lub włamania na jego konto.

- **Podnoszeniu świadomości**

Kontrolowanie zewnętrznych parametrów finansowych i ich zmian w czasie może ostrzegać banki o tym, jak ich klienci i pracownicy będą się zachowywać w przyszłości. Na przykład, nagłe zmiany wskaźników rynkowych mogą prowadzić do zmiany prognoz dotyczących oczekiwanych wycofań (ang. *churn prediction*) lub wysokości wydatków dokonywanych przez klientów. Podobnie można oczekiwać, że nietypowe zdarzenia na rynkach zagranicznych lub nieoczekiwane wahania na rynkach walutowych wpłyną na niektóre działania dużych instytucji prowadzących interesy z bankiem.

- **Wykrywaniu oszustw finansowych**

Nieuczciwe działania ze strony klientów obejmują pranie brudnych pieniędzy, handel nielegalnymi towarami oraz celowe próby wprowadzenia w błąd banku lub innych podmiotów za pomocą wielu transakcji. Analiza sekwencji transakcji w czasie (i porównywanie ich ze wzorcami dla innych kont) może ujawnić anomalie, które wymagają dokładnej analizy i dalszych działań.

Detekcja anomalii znajduje szerokie zastosowanie w cyberbezpieczeństwie. Odpowiednie zabezpieczenie danych firmowych jest niezwykle ważne z punktu widzenia wizerunkowego, ale też prawnego. Brak działań w tym zakresie może prowadzić do wielu naruszeń. Ich skutki nie mają jedynie charakteru finansowego: w 2017 roku aż 40% wszystkich incydentów wiązało się z dłuższymi niż 3 godziny przestojami w prawidłowym funkcjonowaniu spółek. W czasie takich przestojów działalność firmy może zostać całkowicie sparaliżowana lub utrudnione może zostać funkcjonowanie poszczególnych działów. Na ataki i naruszenia jest narażona każda firma. W 2017 roku 44% firm poniosło straty finansowe na skutek ataków, 62% spółek odnotowało zakłócenia i przestoje funkcjonowania, a 21% padło ofiarą zaszyfrowania dysku (tzw. *ransomware*) (Raport PwC, 2018). Wykrywanie anomalii można wykorzystać do:

- detekcji złośliwego oprogramowania,
- wykrywania włamań do sieci,
- wykrywania podejrzanego zachowania pracowników firmy,
- wczesnego wykrywania zagrożeń.

Przegląd zastosowań metod wykrywania anomalii, znajduje się w *Tabeli 6 Zastosowanie metod wykrywania anomalii w cyberbezpieczeństwie i usługach finansowych*.

Tabela 6. Zastosowanie metod wykrywania anomalii w cyberbezpieczeństwie i usługach finansowych.

Nazwa metody	Opis zastosowania	Referencje
Metody statystyczne – Low Pass Filter	Cyberbezpieczeństwo – wykrywanie włamań	(Arvani i Rao, 2014)
	Cyberbezpieczeństwo – wykrywanie fałszywych ataków na kontrolery	(Baba i in. 2018)
One-class SVM	Cyberbezpieczeństwo – wykrywanie złośliwego oprogramowania	(Burnaev i Smolyakov, 2016)
	Cyberbezpieczeństwo – wykrywanie włamania do sieci i ataków cybernetycznych	(Ghanem i in. 2017)
	Bankowość – wykrywanie oszustw finansowych	(Abdelhamid i in., 2014), (Gyam i Abdulai, 2018)
LOF	Cyberbezpieczeństwo – wykrywanie anomalii w ruchu sieciowym	(Hariharan I in., 2019)
	Cyberbezpieczeństwo – wykrywanie nieznanych wcześniej ataków	(Alsmadi I in., 2017)
	Bankowość – wykrywanie oszustw związanych z kartami kredytowymi	(John i Naaz, 2019)
	Bankowość – przeciwdziałanie praniu pieniędzy	(Gao, 2009)
DBSCAN	Cyberbezpieczeństwo – wykorzystanie DBSCAN w wykrywaniu botneta	(Mai i Park, 2016)
	Cyberbezpieczeństwo – wczesne wykrywanie zagrożeń	(Yan i J. Zhang, 2013), (arooq i Otaibi, 2018)
	Cyberbezpieczeństwo – wykrywanie włamań	(Chen i Fei Li, 2011)
	Bankowość – segmentacja klientów	(Zakrzewska i Murlewski, 2005)
	Bankowość – prognozowanie finansowe	(Pavlidis i in. 2007)
Metody oparte na grafach	Cyberbezpieczeństwo – wykrywanie anomalii w sieciach dynamicznych	(Ranshous i in., 2015)
	Cyberbezpieczeństwo – predykcja ataków	(Abraham, Nair, 2015)
	Cyberbezpieczeństwo – detekcja złośliwego oprogramowania	(Xiao i in., 2019)
	Bankowość – wykrywanie oszustw finansowych	(Tarapata i in., 2018)
Metody działające na strumieniach danych	Cyberbezpieczeństwo i bankowość – wykrywanie anomalii	(Ahmad, Purdy, 2016), (Ahmad i in., 2017), (Tan I in., 2011)

Źródło: opracowanie własne.

5.3. Metody wykrywania anomalii zastosowane w rozpoznawaniu botów

Bot jest to skrót od słowa robot, w informatyce oznacza program wykonujący określone działania, mające na celu symulacje żywego człowieka. Naśladuje on różne działania, od kliknięć myszą po wywołanie określonych działań na stronie. Nie jest przygotowany na nieoczekiwane sytuacje, gdyż działa zgodnie z zaprogramowanym algorytmem, co może spowodować liczne załamania w sekwencji wykonywanego kodu.

„Złe" boty (*Bad bots*) poruszające się po stronach stanowią ponad 20% całkowitego ruchu w sieci (Rysunek 23). Przy czym rośnie liczba botów typu *Advanced Persistent Bots* (APBs) wynosząca 73,6% ruchu złych botów. APBs są to programy symulujące ludzkie zachowanie, nawet takie jak ruch myszą. Złe boty mogą być częścią botnetu kierowanego przez botmastera, który zarządza nimi używając kanału komunikacyjnego Command and control (C&C)[17]. Wykorzystuje się je do ataków DDoS, spamowania, phishingu i kradzieży tożsamości (Distil Networks 2019).

„Dobre" boty (*Good bots*), takie jak GoogleBot[18] lub Baidu Spider[19] (boty indeksujące), są pomocnymi wyszukiwarkami, które pomagają dopasować strony internetowe do zapytań. Kolejnym przykładem są chatboty (np. Alexa[20] lub Siri[21]), mające na celu prowadzenie konwersacji z internautą za pośrednictwem algorytmu rozpoznającego język naturalny i odpowiadanie na proste pytania. Takie aktywności stanowią na stronach internetowych 17,5% ruchu (Rysunek 23 Zestawienie ruchu w Internecie w 2018 r. według Distil Networks. Źródło: opracowanie własne). W porównaniu z rokiem 2017 odsetek ten spada (Distil Networks 2019).

5.3.1. Działania złych botów

Na przestrzeni lat boty ewoluowały zmieniając kanały komunikacji ze swoimi twórcami, architekturę, czy sposoby wykradania danych.

Pierwszą generację botów stanowiły proste programy, dokonujące stosunkowo małych szkód (np. Agobot, SDBot, SpyBot). Działały głównie za pośrednictwem usługi IRC[22] wykorzystując scentralizowaną architekturę sieci do wymiany komunikatów między botmasterem a botami. Boty tej generacji korzystały zwykle z niewielkiej liczby adresów IP, więc główną metodą zapobiegania było ich blokowanie.

Boty drugiej generacji (np. Rustock, Zeus, Citadel, Asprox) wykorzystują zdecentralizowaną architekturę i mechanizmy fluxing[23] (Szyling, 2010) oraz zmienne IP (tzw. *fat-flux*) w celu uniknięcia wykrycia. Korzystają też z architektury Peer-2-Peer, w której boty nie kontaktują się z żadnym konkretnym serwerem, ale otrzymują polecenia od innych botów.

Trzecia generacja botów (np. Conficker, Nugache, Storm) potrafi infekować urządzenie bez wiedzy użytkownika, przekształcając je w tzw. komputer Zombie, który staje się częścią sieci botów. Wykorzystują mieszaną architekturę C&C i P2P (opartą na dwóch kategoriach scentralizowanej i zdecentralizowanej) oraz protokół HTTP. Główna metoda wykrywania takich botów, to analiza behawioralna.

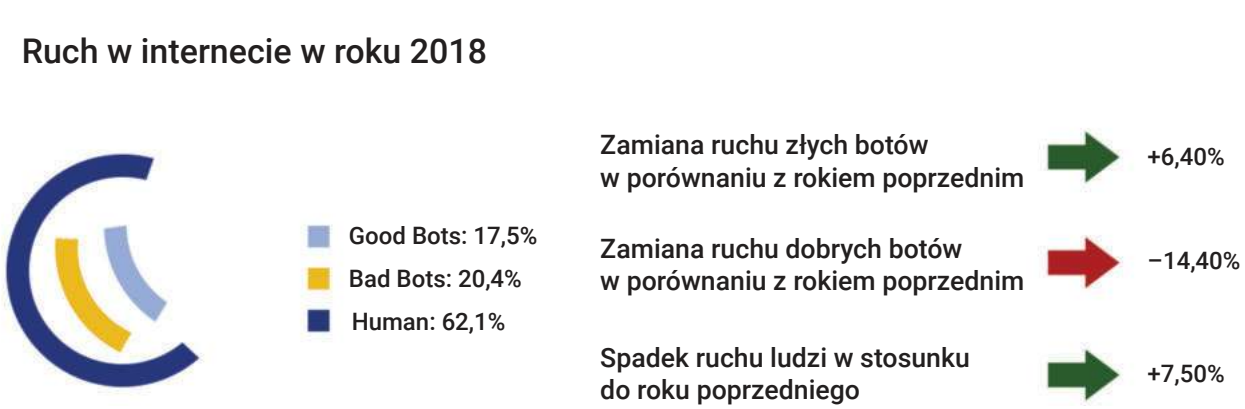
Ostatnia, najbardziej zaawansowana kategoria botów, wykorzystywana jest do przejmowania kont, ataków DDoS i oszustw reklamowych. Boty tego typu imitują zachowanie człowieka, takie jak nieliniowe ruchy myszy. Można je wykryć poprzez tzw. niską analizę behawioralną: ruchy myszy, URL, odcisk cyfrowy, działania klawiatury.

Coraz trudniej jest odróżnić użytkownika od potencjalnego bota, ze względu na wyrafinowanie i szybką adaptację botów do nowych rozwiązań technologicznych. Przykładowo, Advanced Persistent Bots (APBs) mają kilka zaawansowanych możliwości, które pomagają im naśladować ludzkie zachowania. Są znacznie trudniejsze do zidentyfikowania, ponieważ nie są zauważane przez wiele istniejących rozwiązań bezpieczeństwa. Te zaawansowane zachowania obejmują ładowanie zewnętrznych zasobów, manipulowanie plikami cookie oraz automatyzację przeglądarki. Boty te są w stanie również manipulować ładowaniem JavaScriptu oraz wybierać rotacyjne adresy IP[24] (Genovese i in. 2018).

Złe boty są wykorzystywane m.in do:

- Rozsyłania spamu (wiadomości, niezależnie od jej treści, która jest wysyłana do odbiorców, którzy o nią nie prosili) – w sieciach botnetu, botmaster poleca

[17] C&C jest to kanał sterująco-kontrolny służący do wydawania poleceń botom.
[18] GoogleBot: https://support.google.com/webmasters/answer/182072?hl=pl
[19] Baidu Spider: http://www.baiduguide.com/baidu-spider/, http://www.baiduguide.com/baidu-seo-guide/baiduspider-crawl-technology/
[20] Alexa: https://www.amazon.com/b?ie=UTF8&node=17934671011
[21] Siri: https://www.apple.com/siri
[22] IRC – usługa sieciowa, umożliwiająca kontakt klient-server.
[23] Metoda polega na stałej zmianie adresów IP na serwerach DNS (co kilkaset milisekund). Metoda inaczej zwana Fast-Flux w bardzo krótkim czasie pod FQDN (pełną nazwę DNS) podstawić setki różnych adresów IP komputerów z różnych części świata.
[24] Rotacja adresów IP jest to proces dystrybucji przydzielonych adresów IP do odpowiednich urządzeń najczęściej losowo.



Rysunek 23. Zestawienie ruchu w Internecie w 2018 r. według Distil Networks.
Źródło: opracowanie własne na podstawie (Distil Networks 2019)

swoim botom wysyłanie wiadomości typu spamu na zdefiniowaną listę ofiar. Często głównym celem jest phishing – metody techniczne (takie, jak spoofing e-mail[25]), które mają przekonać odbiorców phishingu do ujawnienia swoich osobistych lub profesjonalnych danych poufnych.

- Wypychania danych (*Credential stuffings*) – bot korzystający z bazy skradzionych danych uwierzytelniających konta atakuje wprowadzając (wypychając) je do stron logowania np. na stronach banków. Celem jest przejęcie konta, a wskaźnik powodzenia tego ataku jest wysoki, z uwagi na używanie tego samego hasła uwierzytelniającego przez wiele osób lub w wielu różnych systemach (Chatterjee, 2019).

- Pobieranie danych ze stron internetowych (*Web Scraping Crawlers*) – bot skanuje i wydobywa dane strony, często objęte prawami autorskimi lub znakami towarowymi, a następnie wykorzystuje je w celach konkurencyjnych na własnych stronach internetowych (Zhao, 2017).

- ataki DDoS (*Distributed Denial-of-Service*) – mają zakłócić działanie serwera poprzez zalanie go ruchem internetowym. Bot (często jest to zsynchronizowany atak botów, korzystających z zainfekowanych urządzeń tzw. zombie) wysyła wiele zapytań do serwera, by spowolnić lub zawiesić jego działanie (Hyun i Lee, 2015) (Pompon i Walkowski, 2019).

5.3.2. Przeciwdziałanie złym botom

Boty przeprowadzają różnego typu złożone ataki, dlatego nie można przewidzieć jednego rozwiązania na zatrzymanie ich ekspansji. Można je powstrzymać poprzez wdrożenie kilku warstw ich wykrywania, które dzielą się na dwie główne grupy:

- **Wykrywanie botnetu w oparciu o anomalie sieciowe**, stosując do tego techniki uczenia maszynowego oraz eksploracji danych (*data mining*) w celu odróżnienia i klasyfikacji ruchu sieciowego: zbieranie i znakowanie ruchu botnetowego; generowanie zbioru danych szkoleniowych i testowych z zebranego ruchu; projektowanie jednej lub wielu funkcji; trenowanie ML i modeli data mining za pomocą zestawu danych testowych; testowanie modeli za pomocą zestawu danych testowych; obliczanie wskaźnika wykrywalności i wskaźnika fałszywego alarmu.

- **Wykrywanie botnetu wykorzystując specyficzne cechy botnetu**, za przykład może posłużyć algorytm DGA[26], który jest specyficzny dla botów. Wykrywany za pomocą analizy DNS[27] (Domain Name System)

Do wykrywania botów stosowane są m.in. wymienione poniżej narzędzia.

ADAM (Wykrywanie włamań za pomocą eksploracji danych) jest jednym z podejść z wykorzystaniem eksploracji danych (*data mining*) do rozpoznawania wzorców. Najpierw pozyskuje dane szkoleniowe, które później grupuje na typy ataków, nieznane zachowania oraz fałszywe alarmy. ADAM został zaprojektowany i wdrożony jako system wykrywania anomalii, ale wykorzystuje moduł klasyfikujący podejrzane zdarzenia do fałszywych alarmów lub prawdziwych ataków. ADAM jest unikalny z dwóch powodów. Po pierwsze, wykorzystuje eksplorację danych do budowania własnego profilu reguł normalnego zachowania oraz klasyfikatora, który przesiewa podejrzane działania, klasyfikując je do prawdziwych ataków i fałszywych alarmów. Po drugie, został zaprojektowany w taki sposób, aby można go było używać w czasie

[25] E-mail spoofing – wysyłanie spreparowanych e-maili, podszywających się pod inne źródło, by wyłudzić poufne dane.

[26] Domain generation algorithm – algorytm generujący domeny przez które botmasterzy komunikują się z botami.

[27] System zarządza i dopasowuje nazwy witryn do ich adresów IP np. wpisując 'pl.wikipedia.org' system dobiera IP 91.198.174.192 łącząc nas ze stroną.

rzeczywistym, co zostało osiągnięte dzięki zastosowaniu algorytmów inkrementalnej eksploracji (*incremental mining*), które wykorzystują ruchome okno czasowe w celu wykrycia podejrzanych zdarzeń (Patra, Panda, 2007).

Naiwny Klasyfikator Bayesowski (N*aive Bayesian Classifier*) – szerzej opisany w rozdz. 2.2.6 Naiwny klasyfikator Bayesowski, jest to klasyfikator probabilistyczny, używany jest do klasyfikacji obiektów poprzez szacowanie prawdopodobieństwa ich przynależności do danej grupy.

Gramatyka bezkontekstowa (*Probabilistic Context-Free Grammars*) (Chomsky, 1956) ma postać $G = (N, \Sigma, R, S)$ i jest w postaci normalnej Chomsky'ego (PNC)[28]. Podejście to stosuje się do detekcji DGA, gdzie:

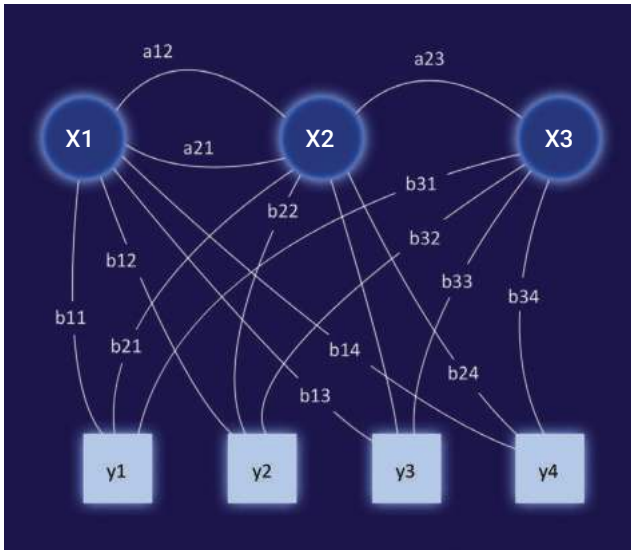
- N jest zbiorem symboli nieterminalnych, które są symbolami zastępczymi dla symboli terminali;
- Σ jest zbiorem symboli końcowych, które są znakami z alfabetów, z których wykonane są łańcuchy;
- R jest zbiorem reguł produkcji w formie $A \rightarrow \alpha$, gdzie $A \in N$ i $\alpha \in (\Sigma \cup N)$, który opisuje rekurencyjny wzór gramatyki;
- S jest symbolem początkowym, a $S \in N$, który jest specjalnym nieterminalnym symbolem pojawiającym się w początkowym łańcuchu generowanym przez gramatykę.

Gramatyka bezkontekstowa probabilistyczna (PCFG), oznaczona przez (G, θ) wykorzystuje wektor prawdopodobieństwa θ do przypisania prawdopodobieństwa każdej z reguł produkcyjnych w R (Fu, 2017). PCFG są naturalnym modelem stochastycznym dla procesów rekurencyjnych (Lu, 2012). PCFG można również opisać jako proces losowego rozgałęzienia, a jego asymptotyczne zachowanie można scharakteryzować poprzez szybkość rozgałęzienia. Szybkość rozgałęzień odnosi się do geometrycznego tempa wzrostu pochodnej zdania.

Test Turinga[29] – najbardziej rozpowszechnioną i znaną formą testu jest CAPTCHA (*Completely Automated Public Turing test to tell Computers and Humans Apart*) stosowana do autoryzowanego logowania, pobierania treści z serwera i innych znaczących działań na stronie. Najczęściej stosowanym zabezpieczeniem CAPTCHA jest rozpoznawanie obrazów (Bala, Saini, 2013) (użytkownik jest zobowiązany wybrać obrazki przedstawiające dany obiekt), rozwiązywanie podstawowych zadań matematycznych (Rysunek 24), określenie wyświetlanego koloru czy wykazywanie podstawowej wiedzy (Rapaport, 2016).

Rysunek 24. Przykładowe zadanie CAPTCHA

Ukryte Modele Markowa (*Hidden Markov Model*, HMM) (Yin i in., 2003) jest procesem podwójnie stochastycznym z procesem stochastycznym, który nie jest obserwowalny, ale może być obserwowany tylko przez inny zestaw procesów stochastycznych, które produkują sekwencję obserwowanych symboli. Jest to narzędzie użytkowe do modelowania informacji o sekwencji. Stany HMM reprezentują pewien nieobserwowalny stan modelowanego systemu. W każdym z tych stanów istnieje pewne prawdopodobieństwo wytworzenia któregokolwiek z obserwowalnych wyjść systemu i oddzielne prawdopodobieństwo wskazujące na prawdopodobieństwo wystąpienia kolejnych stanów. Dzięki różnym rozkładom prawdopodobieństwa wyjściowego w każdym ze stanów



Rysunek 25. HMM zbudowany dla Wikipedii.
Źródło: https://en.wikipedia.org/wiki/Hidden_ Markov_model#/media/File:HiddenMarkovModel.svg

[28] PNC jest to postać gramatyki bezkontekstowej, gdzie małe litery oznaczają symbole terminalne, a duże nieterminalne.
[29] Test Turinga – opracowany przez jednego z twórców informatyki, test ma na celu określić zdolności maszyny do myślenia.

i pozwalając systemowi na zmianę stanów w czasie, model jest w stanie reprezentować sekwencje niestacjonarne (Fu, 2017).

Na Rysunku 25 przedstawiono parametry probabilistyczne ukrytego modelu Markova, gdzie *X* oznacza stany, *y* – możliwe obserwacje, a – prawdopodobieństwo przejścia stanu oraz *b* – prawdopodobieństwo wyjścia.

Masowe blokowanie IP (*Bulk IP blocking*) znanych już „złych" adresów IP[30]. Boty pierwszej generacji znane od dekad, wykorzystują proxy kilku adresów IP do ataków. Adresy te są już powszechnie znane oraz blokowane. Z uwagi na przestarzałą technologię łatwą do wykrycia, nie stanowią znaczącego odsetku.

5.3.3. Zastosowanie detekcji botów w bankowości

Według raportu Distil Networks (Distil Networks 2019) w 2018 na pierwszym miejscu usytuowały się złe boty atakujące sektor finansowy, zajmujące aż 42,2% ruchu na takich stronach. Przegląd metod detekcji botów przedstawia Tabela 7 Zastosowanie metod detekcji botów w cyberbezpieczeństwie i usługach finansowych

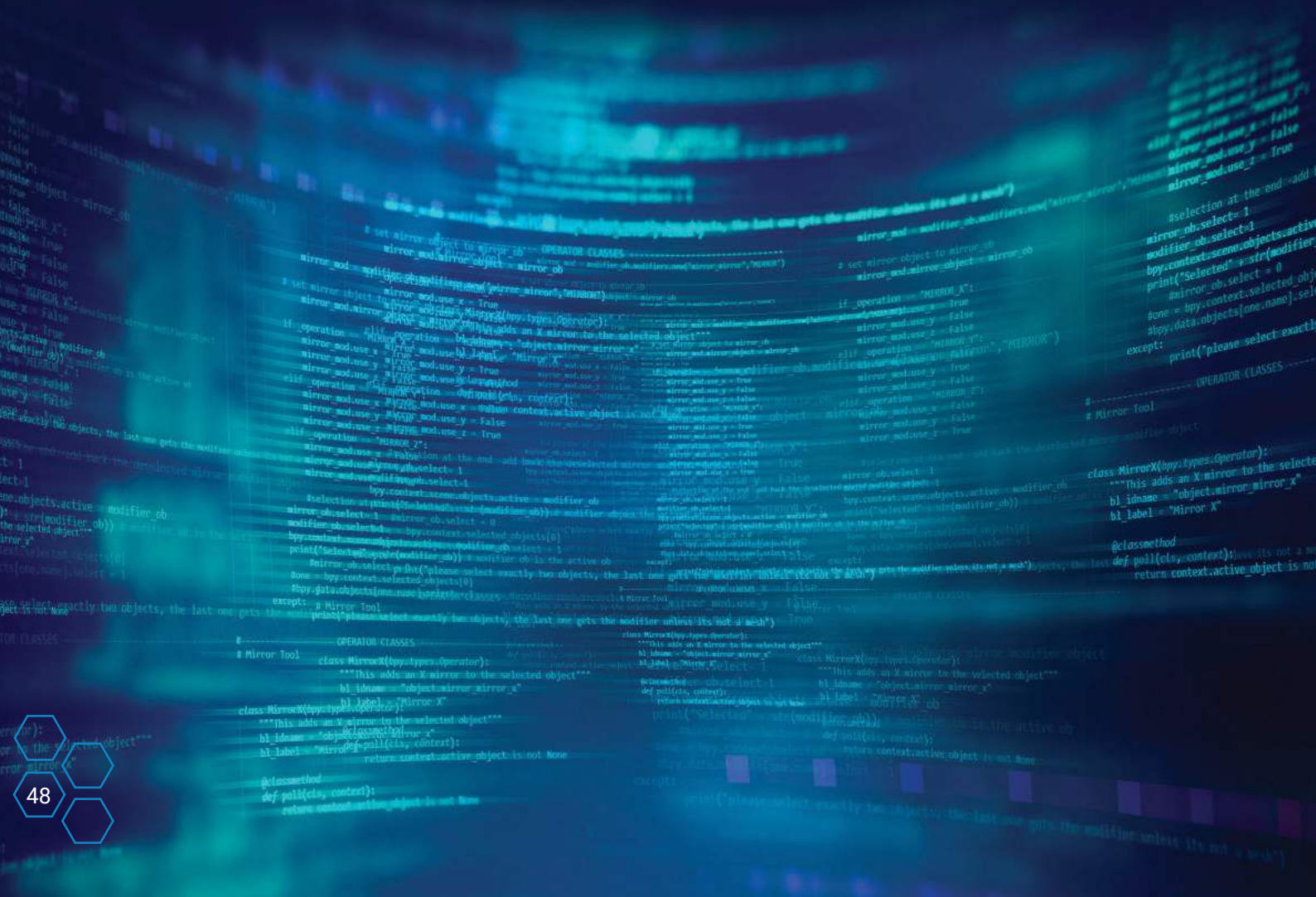
Tabela 7. Zastosowanie metod detekcji botów w cyberbezpieczeństwie i usługach finansowych.

Nazwa Metody/Podejścia	Opis	Zastosowanie	Referencje
ADAM	Metoda wykrywa włamania oraz anomalie w ruchu sieciowym	–	(Patra, Panda, 2007) (Bhambri, 2011)
PCFG	Metoda wykrywa ruch botów w sieci	–	(Lu, 2012) (Booth i R. A. Thompson. 1973)
CAPTCHA	Zapobiega działaniom botów na stronie	IPKO	(Bala, Saini, 2013) (Rapaport, 2016)
HMM	Metoda wykrywa włamania oraz anomalie w ruchu sieciowym	–	(Yin i in., 2003)
Bulk IP blocking	Przeciwdziała znanym już złym botom	Bank Spółdzielczy	(Nachenberg, 2014)

Źródło: opracowanie własne.

[30] Przykładem strony zajmującej się zbieraniem takich adresów jest: https://www.blocklist.de/en/index.html

Wnioski i rekomendacje



We wcześniejszych rozdziałach przestawione zostały najważniejsze metody z dziedziny AI. Choć część z nich jest obecnie z powodzeniem stosowana w cyberbezpieczeństwie, to w dalszych ciągu można zidentyfikować obszary cyberbezpieczeństwa, w których potencjał AI nie jest jeszcze w pełni wykorzystany. Ostatnia dekada przyniosła szybki rozwój metod opartych na AI. Znajdują one zastosowanie w zróżnicowanych problemach związanych z klasyfikacją obiektów, analizą regresji oraz analizach predykcyjnych. Zaprezentowana w raporcie analiza najpopularniejszych metod, przedstawiona w rozdziałach 2 i 3 wykazuje, że w przypadku analizy regresji wykorzystywane są przede wszystkim metody uczenia nadzorowanego (np. regresja liniowa, drzewa decyzyjne, SVM) oraz sieci neuronowe, w klasyfikacji najpopularniejsze są metody analizy skupień, część metod uczenia nadzorowanego (np. lasy losowe) oraz sieci neuronowe, natomiast do predykcji wykorzystuje się przede wszystkim uczenie przez wzmacnianie (np. Q-learning, DQN) oraz sieci neuronowe.

Przedstawione omówienie metod pokazuje, że są one bardzo zróżnicowane i trudno jest wskazać, które będą dawały najlepsze rezultaty w konkretnym zastosowaniu.

Modele analityczne AI są trenowane na zadanym zbiorze danych, uwzględniając ich specyficzną strukturę. Z tego względu modele wytrenowane na danych wyłącznie jednej organizacji (np. pojedynczego banku) niekoniecznie będą bezpośrednio przenaszalne na inne podmioty w obrębie branży lub poza nią. Wytrenowanie modeli na danych z większej liczby podmiotów pozwoli na większą uniwersalność i lepsze dopasowanie modeli do potrzeb sektora. Integracja danych i trenowanie modeli na ich większym zbiorze pozwoli więc na osiągnięcie korzyści skali w obrębie sektora.

Przesłankami wyboru określonej metody będą przede wszystkim charakterystyki zbioru danych, który ma być przetwarzany: jego wolumen, typ danych, możliwość przygotowania zbioru uczącego, itp. Można również wskazać metody, które zostały opracowane z myślą o konkretnym problemie badawczym, np. metody analizy skupień oparte na gęstościach oraz sieci CNN, które bardzo dobrze nadają się do przetwarzania obrazów.

Wybór pomiędzy metodami uczenia nadzorowanego i nienadzorowanego jest uwarunkowany przede wszystkim możliwością pozyskania zbioru uczącego, tj. próbki danych, która będzie stanowiła „wzorzec" rozwiązania (np. obiekty wraz z prawidłową ich klasyfikacją). Pozyskanie takiego zbioru bywa niekiedy niemożliwe – zdarza się tak w przypadku, gdy postawiony problem dotyczy nowego zjawiska lub zgromadzenie potrzebnej liczby obiektów, reprezentujących wszystkie analizowane klasy, będzie zbyt czasochłonne lub kosztowne. Zbiór uczący musi mieć odpowiednio duży wolumen, którego również nie da się określić odgórnie – wymagana ilość danych uczących zależy od zakresu analizy, liczby klas, liczby występujących anomalii, jakości danych, itp. W przypadku danych w formie szeregów czasowych lub wynikających z nowych, nieobserwowanych wcześniej zjawisk, zebranie potrzebnego zbioru danych może trwać nawet kilka lat.

Jakość wyników zwracanych przez metody uczenia nadzorowanego jest uzależniona od jakości danych uczących, jednak są one w większości dobrze przetestowane, łatwe w implementacji i od razu po wytrenowaniu modelu dają trafne rezultaty.

Metody uczenia nienadzorowanego oraz uczenia ze wzmacnianiem mają tę zaletę, że pozwalają na odkrywanie nowych faktów i zjawisk. Ponieważ nie wymagają zbioru uczącego, a zamiast tego budują bazę wiedzy systematycznie, na podstawie własnych wyników, mogą niemal natychmiastowo dostosowywać się do zmian w analizowanym środowisku. Jednakże ich działanie w długim okresie jest mniej przewidywalne, ponieważ ich baza wiedzy może ewoluować na podstawie nowych obserwacji. Metody z tej grupy stopniowo budują bazę wiedzy, więc w początkowym etapie działania zwracane wyniki mogą być niższej jakości, ale powinno się to poprawić wraz z kolejnymi iteracjami algorytmu.

Z przeprowadzonych analiz wynika, że najbardziej uniwersalnymi metodami są obecnie ANN i ich pochodne – nadają się do rozwiązywania zróżnicowanych problemów, dając przy tym dobre rezultaty. Charakteryzują się one jednak istotnymi wadami: przede wszystkim wymagają stosunkowo dużych zbiorów uczących oraz bardzo dużych mocy obliczeniowych, niezbędnych do wytrenowania sieci. Samo pozyskanie zbiorów uczących (wraz z ich kategoryzacją) często stanowi barierę niepozwalającą na zastosowanie ANN – wielkość zbioru uczącego zależy od liczby parametrów sieci i postawionego problemu, ale eksperymenty wskazują, że skuteczne wytrenowanie modeli jest możliwe przy co najmniej kilkuset tysiącach obserwacji. Ponadto, działanie sieci neuronowych jest określane jako „czarna skrzynka", co oznacza, że choć działanie sieci jest deterministyczne, to ścieżka wnioskowania, która doprowadziła do określonego wyniku, może nie być możliwa do odtworzenia przez człowieka. W pewnych specyficznych zastosowaniach związanych z bezpieczeństwem (np. możliwość wykorzystania wyników zwróconych przez sieci jako materiałów dowodowych), może być to istotnym problemem.

Metody oparte na AI stosowane są zarówno do budowania złośliwego oprogramowania, jak i do blokowania działania programów tego typu. Przykładem mogą być boty, opisane w rozdziale 5.3.

AI już w tej chwili jest wykorzystywane na szeroką skalę w cyberbezpieczeństwie, szczególnie w zagadnieniach związanych z wykrywaniem złośliwego oprogramowania, złośliwych zachowań czy wczesnym wykrywaniu zagrożeń. W tych obszarach stosuje się przede wszystkich metody klasyfikujące obiekty.

AI może pomóc we wczesnym wykrywaniu wielu typów fraudów, w tym popełnianych z wykorzystaniem „złego AI", takich jak: 1) impersonacja (przejmowanie tożsamości klientów banku, pokonywanie zabezpieczeń biometrycznych i w konsekwencji kradzież środków), 2) przejmowanie profili behawioralnych klientów w celu inicjowania zaufanej komunikacji z bankiem oraz 3) bombardowanie rynku (generowanie na masową skalę, często rozproszonych zdarzeń, np. wyłudzanie na raz wielkiej ilości małych pożyczek w wielu bankach).

Dużym obszarem, w którym prowadzone są aktualnie liczne badania, jest wykrywanie anomalii w czasie rzeczywistym. Jak pokazano w Rozdziale 5 Wykrywanie anomalii, metody wykrywania anomalii są zróżnicowane i najczęściej oparte na omówionych wcześniej metodach uczenia maszynowego. Zastosowanie znajdują również techniki oparte na grafach oraz dedykowane przetwarzaniu danych strumieniowych. Obecnie wiele instytucji zajmujących się cyberbezpieczeństwem stosuje automatyczne metody wykrywania anomalii – pozwalają one np. zablokować działania wspomnianych wcześniej botów lub niemal natychmiastowo wykryć próby przejęcia tożsamości. Warto jednak odnotować, że tego typu systemy są zwykle dedykowane określonemu podmiotowi lub aplikacji. Co więcej, podmioty z tego samego sektora najczęściej nie wymieniają wyników tych analiz pomiędzy sobą. Biorąc pod uwagę przytoczone wcześniej wymagania dotyczące zbioru danych uczących, jest wysoce prawdopodobne, że integracja danych o typowych i anomalnych obiektach zidentyfikowanych w różnych systemach/podmiotach, może znacząco przyczynić się do poprawy wyników osiąganych przez tego typu metody. W sektorze bankowym ma to szczególne znaczenie, ponieważ przestępcy dysponując bazą skradzionych tożsamości próbują wykorzystać je logując się do systemów różnych banków. Z punktu widzenia pojedynczego banku, takie działanie może wyglądać na zwyczajną (udaną lub nie) próbę logowania, ale jeżeli ta sama kombinacja loginu i hasła jest w krótkich odstępach czasu wywoływana w różnych systemach, może to świadczyć o próbie ataku lub działaniu złośliwego oprogramowania. Innym przykładem jest wykrywanie nowych typów zagrożeń, np. nowym botów. W przypadku, kiedy każdy bank polega jedynie na własnym systemie w celu wykrycia takiego działania, musi on najpierw zgromadzić odpowiednie próbki danych, pozwalające na identyfikację nowego wzorca. Równoczesne gromadzenie danych przez wiele podmiotów pozwoliłoby na szybsze osiągnięcie wymaganego wolumenu danych, a identyfikowany wzorzec anomalii mógłby zostać dystrybuowany do wszystkich podmiotów sektora. Obecnie wyzwaniem jest nie tylko sama detekcja anomalii, ale również opracowanie metod, które byłby w stanie w czasie rzeczywistym przetwarzać duże zasoby danych i na bieżąco generować alerty. Mniejsze banki mogą nie dysponować zasobami pozwalającymi na opracowanie takich rozwiązań, ale dzięki szybkiej dystrybucji informacji o nowych anomaliach oraz współdzieleniu danych uczących, wszystkie podmioty, w tym małe banki, mogłyby poprawić swoje systemy cyberbezpieczeństwa, co z kolei mogłoby wpłynąć na zaufanie klientów do całego sektora bankowego.

Innym obszarem zastosowania AI w bezpieczeństwie, który stosunkowo szybko się rozwija, jest profilowanie behawioralne. Jest ono coraz szerzej wykorzystywane, np. w bezpieczeństwie, marketingu, służy do personalizacji treści. Co więcej, w zasadzie nie ma możliwości pełnego zabezpieczenia się przed pozostawianiem śladów cyfrowych, co powoduje, że metody profilowania behawioralnego są wykorzystywane również w celach przestępczych, np. do automatyzacji ataków phishingowych. Z drugiej strony, profilowanie behawioralne stanowi jeden z najważniejszych trendów w zakresie metod autentykacji użytkownika, gdyż pozwala na uwzględnienie cech użytkownika, zamiast opierania się na weryfikacji na podstawie hasła. Profil behawioralny jest znacznie trudniej „podrobić" lub „wykraść", co powoduje, że jego skuteczna i jednoznaczna identyfikacja może w niedalekiej przyszłości stać się jeszcze ważniejszym zagadnieniem w dziedzinie cyberbezpieczeństwa.

Analiza literatury pozwoliła na zidentyfikowanie rozwiązań opartych na AI, wykorzystywanych w sektorze bankowym i cyberbezpieczeństwie. Zostały one zebrane w *Tabeli 8 Najważniejsze zastosowania metod AI w cyberbezpieczeństwie i usługach finansowych*.

Tabela 8. Najważniejsze zastosowania metod AI w cyberbezpieczeństwie i usługach finansowych.

Zastosowanie	Metody	Referencje
Identyfikacja złośliwego oprogramowania	KNN, regresja logistyczna, drzewa decyzyjne i lasy losowe, klasyfikator Bayesa, SVM, analiza skupień, RNN, CNN, metody grafowe	(Liao i Vemuri, 2002) (Mahdavifar i Ghorbani, 2019) (Markey i Atlasis, 2011) (Shahadat i in., 2017) (Ranjan i Sahoo, 2014) (Babu I in., 2019) (Pascanu i in. 2015) (Vu I in., 2019) (Burnaev i Smolyakov, 2016) (Xiao i in., 2019)
Identyfikacja fałszywych roszczeń wobec ubezpieczyciela	regresja logistyczna, ensemble methods	(Mahdavifar i Ghorbani, 2019) (Xu i in., 2011)
Wykrywanie nowych typów ataków	drzewa decyzyjne i lasy losowe, analiza skupień, LSTM, LOF	(J. H. Wilson, 2009) (Wei I in., 2018) (Filonov i in. 2017) (Alsmadi I in., 2017)
Wykrywanie włamań i systemy IDS	drzewa decyzyjne i lasy losowe, klasyfikator Bayesa, Q-learning, RNN, LSTM, DNN, SVM, DBSCAN, metody grafowe	(Manish i in. 2012) (Li i in. 2017) (Nguyen i Reddi, 2019) (Gulmez, 2019) (Kim i Park, 2019) (Stefanova i Ramachandran, 2018) (Torres i in., 2016) (Babu I in., 2019) (Kim i in. 2016) (Vinayakumar I in, 2017b) (Ghanem i in. 2017) (Yan i J. Zhang, 2013) (Farooq i Otaibi, 2018) (Chen i Fei Li, 2011)(Abraham, Nair, 2015)
Klasyfikacja adresów IP	SVM	(Dangi, 2012)
Segmentacja klientów	ensemble methods, analiza skupień	(Lin i in. 2017) (Wu i Po-Hsuan Chou, 2011) (Zakrzewska i Murlewski, 2005)
Predykcja bankructwa i ryzyka bankowego	ensemble methods, LSTM, DBSCAN	(Mihalovic, 2016) (Li I in. 2018) (Pavlidis i in. 2007)
Detekcja oszustw finansowych	Q-learning, RNN, LSTM, DNN, SVM, LOF, metody grafowe	(El Bouchti i in. 2017) (Babu I in., 2019) (Patel i in. 2019) (Pouramirarsalani i in., 2017) (Heryadi i Warnars, 2017) (Abdelhamid i in., 2014) (Gyam i Abdulai, 2018) (John i Naaz, 2019) (Gao, 2009) (Tarapata i in., 2018)
Wykrywanie wyłudzeń informacji (phishing)	RNN, DNN	(Bahnsen i in. 2017) (Ra i in., 2018)
Wykrywanie anomalii	analiza skupień, LSTM metody statystyczne, SVM, LOF, DBSCAN, metody grafowe, metody strumieniowe	(Riadi I in., 2013) (Vinayakumar i in., 2017a) (Baba i in. 2018) (Arvani i Rao, 2014) (Hariharan I in., 2019) (Mai i Park, 2016) (Ranshous i in., 2015)
Wykrywanie spamu	LSTM	(Jain I in. 2019)
Analiza behawioralna	Q-learning, metody statystyczne, klasyfikator Bayesa, KNN, drzewa decyzyjne, reguły asocjacyjne	(Li i Qiu, 2019) (Pannell i Ashman, 2010) (Chen i in. 2010) (Su i Zhang, 2006) (Xiong i in., 2007) (Pachidi i in. 2014) (Xiang i Gong, 2008) (Tan I in., 2011) (Ahmad i in., 2017) (Ahmad, Purdy, 2016)

Źródło: opracowanie własne.

Rekomendacje

1. Stworzenie centralnych (lub chociaż współdzielonych w ramach grup podmiotów) repozytoriów danych treningowych. Gromadzenie danych treningowych przez wiele podmiotów pozwoli na zgromadzenie większego, bardziej zróżnicowanego zbioru, co umożliwi wytrenowanie modeli do rozpoznawania większej liczby klas obiektów. Zgromadzenia takiego zbioru danych możliwe będzie jedynie poprzez pogłębioną współpracę podmiotów z sektora bankowego.

2. Gromadzenie danych, szczególnie szeregów czasowych zdarzeń, które potencjalnie mogą w niedalekiej przyszłości służyć trenowaniu metod AI. Zbudowanie zbiorów testowych, szczególnie dla szeregów czasowych, jest czasochłonne. Przełoży się to na możliwość szybszego i sprawniejszego wdrożenia nowych metod analitycznych.

3. Gromadzenie danych, z których część będzie stanowiła dane osobowe, musi odbywać się w sposób zgodny z przepisami prawa, przede wszystkim z RODO. Wymaga to poinformowania użytkowników o zakresie gromadzonych danych oraz celach i przetwarzania, jak również uzyskanie ich zgody na takie działania.

4. Choć sama analiza danych o zachowaniu użytkowników jest, przy zachowaniu określonych wymagań i uzyskaniu zgód użytkowników, prawnie i etycznie dopuszczalna, to każdorazowo należy rozważyć kwestie etyczne dla możliwych zastosowań wyników takiej analizy. Dyskryminacja określonych grup klientów lub sprzedaż oparta o techniki manipulacji są uważane za praktyki nieetyczne.

5. Specyfika sektora bankowego, powoduje, że jest on szczególnie często obiektem cyberataków. W sektorze bankowym centralne repozytorium danych treningowych powinno być utrzymywane przez zaufany podmiot, który jednocześnie nie jest bezpośrednią konkurencją dla banków.

6. Stworzenie centralnego repozytorium danych treningowych i/lub systemu cyberbezpieczeństwa oferowanego wszystkim podmiotom sektora bankowego będzie korzystne dla mniejszych podmiotów, które nie mają dużych zasobów (finansowych i technicznych) pozwalających na opracowanie dedykowanego rozwiązania we własnym zakresie. Mniejsze podmioty mogą mieć również problemy w samodzielnym zgromadzeniu wystarczającego zbioru danych uczących, który umożliwiłby skuteczne wytrenowanie modeli analizy danych. Natomiast większe podmioty mogą wykorzystać proponowane rozwiązania jako uzupełnienie własnych narzędzi. Ponadto, poprawa bezpieczeństwa sektora jest korzystna dla wszystkich jego uczestników, ponieważ cyberataki często są przeprowadzane w sposób rozproszony, kierując się na różne podmioty (np. próby zalogowania do systemów różnych banków przy pomocy tej samej, wykradzionej tożsamości).

7. Architektura proponowanego repozytorium danych oraz systemów cyberbezpieczeństwa opartych o takie repozytorium będzie uwarunkowana modelami danych posiadanymi przez podmioty udostępniające dane do repozytorium, dostępną infrastrukturą, wymaganiami funkcjonalnymi i pozafunkcjonalnymi. Ze względu na planowane zastosowanie, tj. cyberbezpieczeństwo sektora bankowego, taka architektura powinna zostać opracowana przy udziale wszystkich zainteresowanych stron. Architektura repozytorium i zbudowanych na nim systemów powinna pozostać poufna.Dystrybucja zidentyfikowanych wzorców anomalii pomiędzy podmiotami o podobnym profilu działalności.

8. Analiza danych z całego sektora lub dla grupy podmiotów pozwoli na uzyskanie dokładniejszych wyników. Ataki w branży finansowej często są rozproszone pomiędzy wiele podmiotów. Złośliwe zachowanie może być niewykrywalne jako anomalia na podstawie danych z jednego podmiotu, a jedynie na podstawie podobieństwa wzorców między różnymi podmiotami.

9. Równoległe stosowanie metod przetwarzających dane historyczne i szukających w nich anomalii i współzależności, jak i metod przetwarzania strumieniowego real-time.

10. Integracja danych z wielu źródeł, również zewnętrznych, może umożliwić odkrycie nowych zależności oraz dokładniejsze predykcje (szczególnie w przypadku sztucznych sieci neuronowych).

11. Istniejące metody oparte na AI są obecnie już wystarczająco dojrzałe, aby mogły być stosowane w cyberbezpieczeństwie. Istnieje wiele systemów, zarówno produkcyjnych, jak i eksperymentalnych, które z sukcesem implementują tego typu metody na potrzeby cyberbezpieczeństwa.

12. Sieci neuronowe są obecnie jednym z najszybciej rozwijających się obszarów, pozwalają na analizę zróżnicowanych typów danych i znajdują zastosowanie w różnych rodzajach problemów (regresja, klasyfikacja, predykcja). Choć wyniki

eksperymentów z wykorzystaniem ANN są obecne również w dziedzinie cyberbezpieczeństwa, to barierą mogą być moce obliczeniowe oraz wolumen danych treningowych, niezbędne do wytrenowania sieci.

13. Zasoby danych, systemy analityczne lub systemy wczesnego powiadamiania o wykrytych anomaliach powinny być udostępnione bankom w sposób umożliwiający weryfikację sposobu ich działania oraz samodzielną integrację z ich wewnętrznymi systemami. Taka integracja może obligować banki do przeprowadzenia audytu bezpieczeństwa informacji, tak więc wszelkie rozwiązania oferowane w sektorze bankowym powinny być zgodne z PN-ISO/IEC 20000-1 i PN-ISO/IEC 20000-2.

14. Rozwiązania związane z cyberbezpieczeństwem powinny być zgodne z wytycznymi Strategii Cyberbezpieczeństwa Rzeczypospolitej Polski na lata 2019-2024 (Uchwała nr 125 Rady Ministrów z dnia 22 października 2019 r. w sprawie Strategii Cyberbezpieczeństwa Rzeczypospolitej Polskiej na lata 2019–2024). Proponowane rozwiązania, tj. współdzielone repozytorium danych uczących oraz rozwiązania analityczne nacelowane na poprawę cyberbezpieczeństwa całego sektora bankowego, wpisują się w cele tej strategii.

Bibliografia

D. Abdelhamid, S. Khaoula i O. Atika. "Automatic bank fraud detection using support vector machines". W: The International Conference on Computing Technology and Information Management (ICCTIM), Dubaj. Society of Digital Information and Wireless Communication, 01.2014. str. 10.

Z. Abdullah i A. R. Hamdan. "Hierarchical clustering algorithms in data mining". W: International Journal of Computer, Electrical, Automation, Control and Information Engineering, Springer, vol. 9(10), 2015. str. 1921-1926.

A. Abraham. "Artificial neural networks". Oklahoma State University, Stillwater, OK, USA, W: Handbook of measuring system design, Wiley, 2005.

S. Abraham i S. Nair. "A predictive framework for cyber security analytics using attack graphs", 2015. URL: https://arxiv.org/abs/1502.01240, dostęp 13.01.2020.

S. Ahmad i S. Purdy. "Real-time anomaly detection for streaming analytics", 2016. URL: https://arxiv.org/abs/1607.02480, dostęp 13.01.2020.

S. Ahmad i in. "Unsupervised real-time anomaly detection for streaming data". W: Neurocomputing, 262, Elsevier, 2017. str. 134-147.

L. Akoglu, H. Tong i D. Koutra. "Graph based anomaly detection and description: a survey". W: Data mining and knowledge discovery vol. 29(3), Springer, 2015. str. 626-688.

M. Alazab. "Profiling and classifying the behavior of malicious codes". W: Journal of Systems and Software, 100, Elsevier, 2015. str. 91-102.

H. Aldawood i G. Skinner. "Contemporary Cyber Security Social Engineering Solutions, Measures, Policies, Tools and Applications: A Critical Appraisal". W: International Journal of Security (IJS) vol. 10(1), CSC, 2019. str. 1.

S. Alla i S. K. Adari. "Beginning Anomaly Detection Using Python-Based Deep Learning". Springer, 2019.

I. M. Alsmadi, G. Karabatis i A. Aleroud. "Information fusion for cyber-security analytics". Springer, 2017.

I. Amit i in. "Machine Learning in Cyber-Security-Problems, Challenges and Data Sets", 2018. URL: https://arxiv.org/abs/1812.07858, dostęp 13.01.2020.

A. Arvani i V. S. Rao. "Cyber security of smart grid systems using intrusion detection methods". W: The International Conference on Computer Security and Digital Investigation (ComSec2014), Kuala Lumpur, Malaysia, ISBN: 978-0-9891305-4-7, 2014. str. 21-28.

I. Assent i in. "Anyout: Anytime outlier detection on streaming data". W: International Conference on Database Systems for Advanced Applications, Springer, Berlin, 04.2012. str. 228-242.

Atlantic Council. Risk Nexus. Overcome by cyber risks? Economic benefits and costs of alternate cyber futures. Tech. rep. Atlantic Council, 2018, URL: https://www.atlanticcouncil.org/in-depth-research-reports/report/risk-nexus-overcome-by-cyber-risks-economic-benefits-and-costs-of-alternate-cyber-futures/, dostęp 13.01.2020.

T. O. Ayodele. "Types of machine learning algorithms". W: New advances in machine learning. IntechOpen, UK, 2010.

R. Baba, K. Kogiso i M. Kishida. "Detection method of controller falsification attacks against encrypted control system". W: Proc. of the SICE Annual Conference, Japan, 2018. str. 244-248.

M. H. Babu, R. Vinayakumar, K. P. Soman. "RNNSecureNet: Recurrent neural networks for Cyber security use-cases". W: arXiv preprint arXiv:1901.04281, 2019.

A. C. Bahnsen i in. "Classifying phishing URLs using recurrent neural networks". W:

APWG Symposium on Electronic Crime Research (IEEE), USA, 04.2017. str. 1-8.

A. Bala, B. S. Saini. A Review of Bot Protection using CAPTCHA for Web Security. W: IOSR Journal of Computer Engineering, vol. 8(6), e-ISSN: 2278-0661, p– ISSN: 2278-8727, International Organization Of Scientific Research, 01.2013.

T. Bartłomowicz i in. "Filtrowanie zbioru ofert nieruchomości z wykorzystaniem informacji o preferencjach". W: Warsaw School of Economics, Collegium of Economic Analysis, vol. 23, 2011. str. 25-36.

V. Behzadan i A. Munir. "Vulnerability of deep reinforcement learning to policy induction attacks". W: International Conference on Machine Learning and Data Mining in Pattern Recognition, Springer, 2017. str. 262-275.

T. Berg i in. "On the rise of FinTechs Credit scoring using digital footprints". Tech. rep. National Bureau of Economic Research, 2018, URL: https://www.nber.org/papers/w24551.pdf, dostęp 13.01.2020.

D. S. Berman i in. "A survey of deep learning methods for cyber security". W: Information vol 10(4), 2019. str. 122. DOI: https://doi.org/10.3390/info10040122

K. Bhageshpur. "Data Is The New Oil – And That's A Good Thing", 11.2015. URL: https: //www.forbes.com/sites/forbestechcouncil/2019/11/15/data-is-the-new-oil-andthats-a-good-thing/#26f190277304, dostęp 13.01.2020.

V. Bhambri. "Application of Data Mining in Banking Sector", IJCST, vol. 2(2), 2011, ISSN: 0976-8491, UoCJG, URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.448.1706&rep=rep1&type=pdf, dostęp 13.01.2020.

M. Bianchini i F. Scarselli. "On the complexity of neural network classifiers: A comparison between shallow and deep architectures". W: IEEE transactions on neural networks and learning systems vol. 25(8), 2014. str. 1553-1565.

T. L. Booth i R. A. Thompson. "Applying probability measures to abstract languages". W: IEEE transactions on Computers, vol. 100(5), 1973. IEEE, str. 442-450.

M. PS. Brown i in. "Knowledge-based analysis of microarray gene expression data by using support vector machines". W: Proceedings of the National Academy of Sciences vol. 97(1), National Academy of Sciences, 2000. str. 262-267.

E. Burnaev i D. Smolyakov. "One-class SVM with privileged information and its application to malware detection". W: 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW). IEEE. 2016. str. 273-280.

M. A. Carreira-Perpinan. "Continuous latent variable models for dimensionality reduction and sequential data reconstruction". PhD thesis, University of Sheffield, 2001.

S. E. Chandy i in. "Cyberattack Detection Using Deep Generative Models with Variational Inference". W: Journal of Water Resources Planning and Management vol. 145(2), American Society of Civil Engineers, 2018. str. 04018093.

R. Chatterjee, T. Ristenpart, B. Pal, T. Daniel. "Beyond Credential Stuffng: Password Similarity Models using Neural Networks". IEEE, 2019.

J. Chen i in. "Short and tweet: experiments on recommending content from information streams". W: Proceedings of the SIGCHI conference on human factors in computing systems, ACM, 2010. str. 1185-1194.

M. Chen, A. A. Ghorbani i in. "A Survey on User Pro ling Model for Anomaly Detection in Cyberspace". W: Journal of Cyber Security and Mobility vol. 8(1), River Publishers, 2019. str. 75-112.

R. Chen i in. "A Deep Learning Framework for Joint Image Restoration and Recognition". W: Circuits, Systems, and Signal Processing, Springer, 2019. str. 1-20.

T. Chen i in. "Adversarial attack and defense in reinforcement learning-from AI security view". W: Cybersecurity vol. 2(1), Springer, 2019. str. 11.

Z. Chen i Y. Fei Li. "Anomaly detection based on enhanced DBScan algorithm". W: Procedia Engineering 15, Elsevier, 2011. str. 178-182.

D. Chicco, P. Sadowski i P. Baldi. "Deep autoencoder neural networks for gene ontology annotation predictions". W: Proceedings of the 5th ACM conference on bioinformatics, computational biology, and health informatics, ACM, 2014. str. 533-540.

I. Chomiak-Orsa, A. Rot i B. Blaicke. "Artificial Intelligence in Cybersecurity: The Use of AI Along the Cyber Kill Chain". W: International Conference on Computational Collective Intelligence, Springer, 2019. str. 406-416.

N. Chomsky. "Three models for the description of language. Information Theory". W: IRE Transactions on, vol. 2(3), IEEE, 09.1956.

D. Ciresan i in. “Multi-column deep neural network for traffic sign classification”. W: Neural networks vol. 32, Elsevier, 2012, str. 333-338.

Q. V. Dang. “Reinforcement Learning in Stock Trading”. W: International Conference on Computer Science, Applied Mathematics and Applications. Springer, 2019, str. 311-322.

C. S. Dangi, R. Gupta i G. S. Chandel. “Cyber Security Approach in Web Application using SVM”. W: International Journal of Computer Applications vol. 57(20), Citeseer, 2012.

L. Deng, D. Yu i in. “Deep learning: methods and applications”. W: Foundations and Trends in Signal Processing, vol. 7(3-4), now publishers inc., 2014, str. 197-387.

D. M. Diab i in. “Anomaly Detection Using Dynamic Time Warping”. W: 2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Sierpień, 2019, str. 193-198. DOI: 10.1109/CSE/EUC.2019.00045, dostęp 13.01.2020.

T. G. Dietterich. “Ensemble methods in machine learning”. W: International workshop on multiple classifier systems, Springer, 2000. str. 1-15.

Distil Networks. Bad Bot Report 2019: The Bot Arms Race Continues. Tech. rep. Distil networks, 2019, dostępny online: https://resources.distilnetworks.com/white-paper-reports/bad-bot-report-2019, dostęp 13.01.2020.

H. Drucker i in. “Boosting and other ensemble methods”. W: Neural Computation, vol 6(6), MIT Press, 1994. str. 1289-1301.

A. Duraj i Ł. Chomątek. “Detection of outliers in data streams using grouping methods”. W: Przegląd Elektrotechniczny vol. 2019(2), Politechnika Łódzka, ISSN 0033-2097, 2019. Str. 85-87.

P. Eckersley. “How unique is your web browser?” W: International Symposium on Privacy Enhancing Technologies Symposium. Springer, 2010. str. 1-18.

The Economist. “The world’s most valuable resource is no longer oil, but data.” May 6, 2017. URL: https://www.economist.com/leaders/2017/05/06/the-worlds-most-valuable-resource-is-nolonger-oil-but-data, dostęp 13.01.2020.

A. El Bouchti i in. “Fraud detection in banking using deep reinforcement learning”. W: 2017 Seventh International Conference on Innovative Computing Technology (INTECH). IEEE, 2017. str. 58-63.

A. M. Elkahky, Y. Song i X. He. “A multi-view deep learning approach for cross domain user modeling in recommendation systems”. W: Proceedings of the 24th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, IW3C, 05.2015, str. 278-288.

B. Eubanks. “Artificial Intelligence for HR: Use AI to Support and Develop a Successful Workforce”. Kogan Page, 2018. ISBN: 9780749483814. URL: https://books.google.pl/books?id= XCaAswEACAAJ., dostęp 13.01.2020.

European Payment Council. “2018 Payment Threats and Fraud Trends Report. Tech. rep. European Payment Council”, 2018. URL: https://www.europeanpaymentscouncil.eu/documentlibrary/guidance-documents/2018-payment-threats-and-fraud-trends-report, dostęp 13.01.2020.

J. J. Faraway. “Practical regression and ANOVA using R”. Opublikowane online: https://people.bath.ac.uk/jjf23/book/pra.pdf, 2002, dostęp 13.01.2020.

H. M. Farooq i N. M. Otaibi. “Optimal Machine Learning Algorithms for Cyber Threat Detection”. W: 2018 UKSim-AMSS 20th International Conference on Computer Modelling and Simulation (UKSim). IEEE, 2018. str. 32-37.

P. Filonov, F. Kitashov i A. Lavrentyev. “Rnn-based early cyber-attack detection for the tennessee eastman process”. W: arXiv preprint arXiv:1709.02232, 2017.

F. Foroughi i P. Luksch. “Observation Measures to Profile User Security Behaviour”. W: 2018 International Conference on Cyber Security and Protection of Digital Services (Cyber Security). IEEE, 2018, str. 1-6.

V. Francois-Lavet i in. “An introduction to deep reinforcement learning.”. W: Foundations and Trends R in Machine Learning 11.3-4, Now Foundations and Trends, 2018. str. 219-354.

Y. Fu. “Using Botnet Technologies to Counteract NetworkTraffic Analysis”. TigerPrints. Clemson University, 2017.

N. Ganesan i in. “Application of neural networks in diagnosing cancer disease using demographic data”. International Journal of Computer Applications vol. (1.26), 2010. str. 76-85.

Z. Gao. "Application of cluster-based local outlier factor algorithm in anti-money laundering". W: 2009 International Conference on Management and Service Science. IEEE, 2009. str. 1-4.

A. Genovese i in. "From Database to Cyber Security: Essays Dedicated to Sushil Jajodia on the Occasion of His 70th Birthday". Ed. by P. Samarati, I. Ray, and I. Ray. Springer, 30 lis 2018.

K. Ghanem i in. "Support vector machine for network intrusion and cyber-attack detection". W: 2017 Sensor Signal Processing for Defence Conference (SSPD). IEEE, 2017. str. 1-5.

M. Goldstein i S. Uchida. "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data". PloS one 11.4, 2016, e0152173.

I. Goodfellow, Y. Bengio i A. Courville. "Deep learning". MIT press, 2016.

N. Grover. "A study of various fuzzy clustering algorithms". W: International Journal of Engineering Research vol. (3.3), ISSN 2319-6890, 2014. str. 177-181.

S. Guha i in. "Robust random cut forest based anomaly detection on streams". W: International conference on machine learning, PMLR, 2016. str. 2712-2721.

H. G. Gulmez. "A deep reinforcement learning approach to network intrusion detection". Middle East Technical University, Master Thesis, 2019.

N. K. Gyam i J. D. Abdulai. "Bank Fraud Detection Using Support Vector Machine". W: 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). IEEE, 2018. str. 37-41.

S. Halder i S. Ozdemir. "Hands-on machine learning for Cyber Security". Packt publishing, 2018.

J. Han, M. Kamber i J. Pei. "Data mining concepts and techniques third edition". W: The Morgan Kaufmann Series in Data Management Systems, Elsevier, 2011. str. 83-124.

A. Hannun i in. "Deep speech: Scaling up end-to -end speech recognition". W: arXiv preprint arXiv:1412.5567, 2014.

A. Hariharan, A. Gupta, i T. Pal. "CAMLPAD: Cybersecurity Autonomous Machine Learning Platform for Anomaly Detection". W: arXiv preprint arXiv:1907.10442, 2019.

S. Haykin. "Neural Networks and Learning Machines" 3rd edition. Pearson Education India, 2010.

D. O. Hebb. "Organization of behavior". Wiley, New York, 1949.

M. Henzinger. "The State of the Art in Dynamic Graph Algorithms". W: International Conference on Current Trends in Theory and Practice of Informatics. Springer, 2018. str. 40-44.

Y. Heryadi i H. L. H. S. Warnars. "Learning temporal representation of transaction amount for fraudulent transaction recognition using CNN, Stacked LSTM, and CNN-LSTM". W: IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), IEEE, 2017. str. 84-89.

V. Hodge i J. Austin. "A survey of outlier detection methodologies". W: Artificial intelligence review vol. (22.2), Springer, 2004. str. 85-126.

P. Hyun i J. Lee "ZombieCoin: Powering Next-Generation Botnetswith Bitcoin". W: Financial Cryptography and Data Security: FC 2015 International Workshops. Springer 2015.

International Monetary Fund. Estimating Cyber Risk for the Financial Sector. Tech. rep. IMF. 2018,URL: https://blogs.imf.org/2018/06/22/estimating-cyber-risk-for-thefinancial-sector, dostęp 13.01.2020.

A. K. Jain, J. Mao i K. M. Mohiuddin. "Artificial neural networks: A tutorial". W: Computer vol. (29.3), IEEE, 1996. str. 31-44.

G. Jain, M. Sharma i B. Agarwal. "Optimizing semantic LSTM for spam detection". W: International Journal of Information Technology vol. (11.2), Springer, 2019. str. 239-250.

S. Janarthanam. "Hands-on chatbots and conversational UI development: Build chatbots and voice user interfaces with Chatfuel, Dialog ow, Microsoft Bot Framework, Twilio, and Alexa Skills". Packt Publishing Ltd, 2017.

N. Jiang, A. Kulesza i S. Singh. "Abstraction selection in model-based reinforcement learning". W: International Conference on Machine Learning, JMLR. org, 2015. str. 179-188.

H. John i S. Naaz. "Credit Card Fraud Detection using Local Outlier Factor and Isolation Forest". W: International Journal of Computer Sciences and Engineering, vol. 7(4), str. 1060-1064 2019.

A. Jovic, K. Brkic i N. Bogunovic." A review of feature selection methods with applications". W: 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). IEEE, 2015. str. 1200-1205.

I. Katsov. "Introduction to Algorithmic Marketing: Artificial Intelligence for Marketing Operations", Ilia Katcov, ISBN-13: 978-0692989043, 2017.

Ch. Kim i J. Park. "Designing online network intrusion detection using deep autoencoder Q-learning". W: Computers & Electrical Engineering vol. 79, Elsevier, 2019, DOI: 10.1016/j.compeleceng.2019.106460.

J. Kim i in. "Long short term memory recurrent neural network classifier for intrusion detection". W: 2016 International Conference on Platform Technology and Service (PlatCon). IEEE, 2016, str. 1-5.

A. Kirschenbaum i in. "Airport security: An ethnographic study." W: Journal of Air Transport Management, vol. 18, Elsevier, 2012. str. 68-73. DOI: 10.1016/j.jairtraman.2011.10.002.

A. Kleiner, P. Nicholas i K. Sullivan. "Linking cybersecurity policy and performance". W: Microsoft Trustworthy Computing, Microsoft Corporation, vol. (1.1), 2014

V. R. Konda i J. N. Tsitsiklis. "Actor-critic algorithms". W: Advances in neural information processing systems, 2000. str. 1008-1014.

M. Koziarski, K. Kwater i M. Woźniak. "Wykorzystywanie programów uczenia w głębokim uczeniu przez wzmacnianie. O istocie rozpoczynania od rzeczy małych". W: Edukacja-Technika-Informatyka, vol. 9(2), Uniwersytet Rzeszowski 2018, str. 220-226.

H. Kriegel i in. "Density-based clustering". W: WIREs Data Mining and Knowledge Discovery 1(3), 2011. str. 231-240.

D. Kriesel. "A brief introduction on neural networks", opublikowany onlline: www.dkriesel.com, 2007. Dostęp 13.01.2020.

C. Li i M. Qiu. "Reinforcement Learning for Cyber-Physical Systems: with Cybersecurity Case Studies". Chapman and Hall/CRC, 2019.

J. Li i C. Manikopoulos. "The application of a low pass Iter in anomaly network intrusion detection". W: Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, 2004. IEEE. 2004. str. 265-271.

N. Li. "Uncovering Interesting Attributed Anomalies in Large Graphs". University of California, Santa Barbara, PhD Thesis, 2013.

X. Li, X. Long, G. Sun, G. Yang and H. Li, "Overdue Prediction of Bank Loans Based on LSTM-SVM". W: 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI). IEEE, 2018. str. 1859-1863.

Y. Li i in. "Detection, classfication and characterization of android malware using api data dependency". W: International Conference on Security and Privacy in Communication Systems. Springer, 2015. str. 23-40.

Y. Li i H. Wu. "A clustering method based on K-means algorithm". W: Elsevier. vol 25, 2012. str. 1104-1109.

Y. Liao i VR. Vemuri. "Use of k-nearest neighbor classfier for intrusion detection". W: Elsevier vol 21(5), 2002. str. 439-448.

W. Lin i in. "An ensemble random forest algorithm for insurance big data analysis". W: Ieee Access 5, 2017. str. 16568-16575.

C. Lu. "Network Traffic Analysis Using Stochastic Grammars". Clemson, SC, USA, 2012.

L. Maaten i G. Hinton. "Visualizing data using t-SNE". W: Journal of machine learning research, 11.2008. str. 2579-2605. URL: http://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf.

S. Mahdavifar i A. Ghorbani. "Application of deep learning to cybersecurity: A survey". W: Neurocomputing 347, 2019. str. 149-176.

L. Mai i M. Park. "A comparison of clustering algorithms for botnet detection based on network flow". W: 2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN). IEEE, 2016. str. 667-669.

K. Manish, M. Hanumanthappa i T. V. K. Suresh "Intrusion detection system using decision tree algorithm". W: 2012 IEEE 14th international conference on communication technology. IEEE, 2012. str. 629-634.

S. Manju i M. Punithavalli. “An analysis of Q-learning algorithms with strategies of reward function”. W: JCSE VOL. 3(2), 2011. str. 814-820.

J. Markey i A. Atlasis. “Using decision tree analysis for intrusion detection: a how-to guide”. W: SANS Institute InfoSec Reading Room, 2011.

T. Matiisen. “Demystifying deep reinforcement learning”. W: Computational Neuroscience LAB, 2015. Opublikowane online: https://neuro.cs.ut.ee/demystifying-deep-reinforcement-learning/, dostęp 13.01.2020.

W. S. McCulloch i W. Pitts. “A logical calculus of the ideas immanent in nervous activity”. W: The bulletin of mathematical biophysics 5.4, 1943. str. 115-133.

M. Mihalovic. “Performance comparison of multiple discriminant analysis and logit models in bankruptcy prediction”. Economics and Sociology, vol 9(4), 2016. str. 101-118. DOI: 10.14254/2071-789X.2016/9-4/6

V. Mnih i in. “Human-level control through deep reinforcement learning”. W: Nature, vol 518(7540) Nature Publishing Group, 2015. str. 529–533 DOI: 10.1038/nature14236

A. Mohamed, G. Dahl, i G. Hinton. “Deep belief networks for phone recognition”. W: NIPS workshop on deep learning for speech recognition and related applications. vol. 1(9). Neural Information ProcessingSystem Foundation, Inc, 2009.

C. K. Mohan i K. G. Mehrotra. “Anomaly detection in banking operations”. W: IDRBT Journal of Banking Technology, IDRBT. 2017. str. 16-18.

A. Moreira, M. Y. Santos, i S. Carneiro. “Density-based clustering algorithms DBSCAN i SNN”. University of Minho-Portugal, 2005.

C. S. Nachenberg. “IP-based blocking of malware”. U.S. Patent No. 8,756,691. 17 Jun. 2014

S. Naval, V. Laxmi, N. Gupta, M. S. Gaur, M. Rajarajan. “Exploring Worm Behaviors using DTW”. W: Proceedings of the 7th International Conference on Security of Information and Networks (SIN ’14). Association for Computing Machinery, New York, NY, USA, str. 379–384. 2014. DOI:https://doi.org/10.1145/2659651.2659737

T. T. Nguyen i V. J. Reddi. “Deep Reinforcement Learning for Cyber Security”. W: CoRR abs/1906.05799, 2019. arXiv: 1906.05799. URL: http://arxiv.org/abs/1906.05799.

D. Nicolls. “Cybersecurity Threats Facing Financial Services”. URL: https://www.jumio.com/5-cybersecurity-threats-financial-services/, 2018, dostęp 13.01.2020.

K. Oh i in. “Question Understanding Based on Sentence Embedding on Dialog Systems for Banking Service”. W: 2019 IEEE International Conference on Big Data and Smart Computing (BigComp). IEEE. 2019. str. 1-3.

A. Ortigosa, R. M. Carro, i J. I. Quiroga. “Predicting User Personality by Mining Social Interactions in Facebook”. W: J. Comput. Syst. Sci. 80.1, 02.2014. str. 57-71. issn: 0022-0000. URL: 10.1016/j.jcss.2013.03.008. URL: http://dx.doi.org/10.1016/j.jcss.2013. 03.008.

W. Owsiak i A. Owsiak. “Rozszerzenie algebry algorytmów”. W: Pomiary Automatyka Kontrola vol. 56, Stowarzyszenie Inżynierów i Techników Mechaników Polskich, 2010. str. 184-188.

S. Pachidi, M. Spruit, i I. Weerd. “Understanding users’ behavior with software operation data mining”. W: Computers in Human Behavior 30, 2014, str. 583-594.

H. H. Pajouh i in. “A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks”. W: IEEE Transactions on Emerging Topics in Computing, 2016.

M. Panda i M. R. Patra. “Network intrusion detection using Naïve Bayes”. W: International journal of computer science and network security 7.12, 2007. str. 258-263.

G. Pannell i H. Ashman. “User modelling for exclusion and anomaly detection: a behavioural intrusion detection system”. W: International Conference on User Modeling, Adaptation, i Personalization. Springer, 2010. str. 207-218.

J. D. Parmar i J. T. Patel. “Anomaly Detection in Data Mining: A Review”. W: International Journal of Advanced Research in Computer Science and Software Engineering, vol. 7.4, 2017.

R. Pascanu i in. “Malware classification with recurrent networks”. W: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015. str. 1916-1920.

A. Pascual, K. Marchini, S. Miller. "2018 Identity Fraud: Fraud Enters a New Era of Complexity". 2018, URL: https://www.javelinstrategy.com/coverage-area/2018-identity-fraud-fraud-enters-new-era-complexity, dostęp 13.01.2020

Y. Patel, K. Ouazzane, V. Vassilev, J. Li, "Remote banking fraud detection framework using sequence learners". W: Journal of Internet Banking and Commerce vol. 24.1, ISSN: 1204-5357, 2019. str. 1-31.

T. R. Patil, S. Sherekar "Performance analysis of Naive Bayes and J48 classification algorithm for data classification". W: International Journal of Computer Science and Applications vol. 6(2) 2013. str. 256-261 ISSN: 0974-1011.

M. R. Patra, M. Panda. "Network Intrusion Detection Using Naive Bayes". W: International Journal of Computer Science and Network Security, IJCSNS, 2007.

N. G. Pavlidis, V. P. Plagianakos, D. K. Tasoulisand, M. N. Vrahatis "Financial forecasting through unsupervised clustering and neural networks". Operational Research vol. 6(2), Springer, 2006. str. 103-127.

J. Pociecha i in. "Statystyczne metody prognozowania bankructwa w zmieniającej się koniunkturze gospodarczej". Fundacja Uniwersytetu Ekonomicznego w Krakowie, Kraków 2014.

R. Pompon i D. Walkowski. "Good Bots, Bad Bots, and What You Can Do About Both.", 01.03.2019, URL: https://www.f5.com/labs/articles/threat-intelligence/good-bots--bad-bots--and-what-you-can-do-about-both, dostęp 13.01.2020.

A. Pouramirarsalani, M. Khalilian, i A. Nikravanshalmani. "Fraud detection in E-banking by using the hybrid feature selection and evolutionary algorithms." W: IJCSNS 17.8, 2017. str. 271-279.

Raport PwC. Cyber-ruletka po polsku. "Dlaczego firrmy w walce z cyberprzestępcami liczą na szczęście." W: Warszawa [online], URL: https://www.pwc.pl/pl/pdf/cyber-ruletka-po-polsku-raport-pwc-gsiss-2018.pdf, 2018.

V. Ra i in. "DeepAnti-PhishNet: Applying deep neural networks for phishing email detection". W: Proc. 1st AntiPhishing Shared Pilot 4th ACM Int. Workshop Secur. Privacy Anal.(IWSPA). Tempe, AZ, USA, 2018. str. 1-11.

R. Ranjan i G. Sahoo. "A new clustering approach for anomaly intrusion detection.", 2014, opublikowane URL: https://arxiv.org/abs/1404.2772, dostęp 13.01.2020.

S. Ranshous i in. "Anomaly detection in dynamic networks: a survey". Wiley Periodicals, Inc. vol 7(3), 2015. str. 223-247.

S. Rapaport, E. Azaria, O. Schwartz. "Distinguish valid users from bots, ocrs and third party solvers when presenting captcha", Stany Zjednoczone, Patent Nr. 9,501,651. 11.2016.

M. V. Reddy, M. Vivekananda i R. Satish. "Divisive Hierarchical Clustering with K-means and Agglomerative Hierarchical Clustering". W: International Journal of Computer Science Trends and Technology, vol. 5.5, 2017. str. 6-11.

A. Riad, I. Elhenawy, A. Hassan, N. Awadallah "Visualize network anomaly detection by using k-means clustering algorithm". W: International Journal of Computer Networks & Communications, vol. 5(5), 2013. str. 195-208.

I. Riadi, J. E. Istiyanto, A. Ashari, i in. "Log analysis techniques using clustering in network forensic", 2013 opublikowane https://arxiv.org/abs/1307.0072, dostęp 13.01.2020

B. D. Ripley. "Pattern recognition and neural networks". Cambridge university press, 2007.

L. Rokach i O. Maimon. "Clustering methods". W: Data mining and knowledge discovery handbook. Springer, 2005, str. 321-352.

D. E. Rumelhart, G. E. Hinton i R. J. Williams. "Learning representations by back-propagating errors". W: Nature 323, Nature Publishing Group, 1986. str. 533-536.

R. Sathya i A. Abraham. "Comparison of supervised and unsupervised learning algorithms for pattern classification". W: International Journal of Advanced Research in Artificial Intelligence, vol. 2(2), The Science and Information (SAI) Organization, 2013, str. 34-38.

S. M. Savaresi i in. "Cluster selection in divisive clustering algorithms". W: Proceedings of the 2002 SIAM International Conference on Data Mining". SIAM, 2002. str. 299-314.

A. Senf, X. Chen i A. Zhang. "Comparison of one-class SVM and two-class SVM for fold recognition". W: International Conference on Neural Information Processing. Springer, 2006. str. 140-149.

G. Seni i J. F. Elder. "Ensemble methods in data mining: improving accuracy through combining predictions". W: Synthesis lectures on data mining and knowledge discovery, red. J. Han, L. Getoor, W. Wang, J. Gehrke, R. Grossman, vol. 2(1), Morgan & Claypool Publishers, 2010. str. 1-126.

N. Shahadat i in. "Experimental analysis of data mining application for intrusion detection with feature reduction". W: 2017 International Conference on Electrical, Computer and Communication Engineering (ECCE). IEEE, 2017. str. 209-216.

S. Singh Rathore, S. Kumar. "A Decision Tree Regression based Approach for the Number of Software Faults Prediction". SIGSOFT Software Engineering Notes vol. 41(1), February 2016, 1–6. DOI:https://doi.org/10.1145/2853073.2853083

R. State, J. Charlier, G. Ormazabal, J. Hilger i in. "MQLV: Optimal Policy of Money Management in Retail Banking with Q-Learning". W: Proceedings of the Fourth Workshop on Mining Data for Financial applications (MIDAS 2019) co-located with the 2019 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD 2019), Würzburg, Germany, Springer, 2019.

Z. Stefanova i K. Ramachandran. "Off-Policy Q-learning Technique for Intrusion Response in Network Security". W: International Journal of Computer and Information Engineering, vol. 12(4), World Academy of Science, Engineering and Technology, 2018. str. 262-268.

Z. Stęgowski. "Sztuczne sieci neuronowe". Wydawnictwa Naukowo-Techniczne, Kraków, 2004.

J. Su i H. Zhang. "Full Bayesian network classifiers". W: Proceedings of the 23rd international conference on Machine learning. ACM, 2006. str. 897-904.

R. S. Sutton i A. G. Barto. "Reinforcement learning: An introduction". MIT press, 2018.

C. Szepesvari. "Algorithms for reinforcement learning". W: Synthesis lectures on artificial intelligence and machine learning, vol. 4(1), Morgan & Claypool Publishers, 2010. str. 1-103.

K. Szyling. "Systemy Agentowe-Sieci BotNet", 2010. URL: http://w-files.pl/wp-content/uploads/2010/05/botnet.pdf, dostęp 13.01.2020.

K. Szymielewicz. „Śledzenie i profilowanie w sieci: W czym problem? Co się zmieni w prawie? Jak może wyglądać przyszłość?", Fundacja Panoptykon, wrzesień 2017, opublikowane online: https://panoptykon.org/sites/default/files/publikacje/sledzenie_i_profilowanie_w_sieci_scenariusze_po_reformie_ue_wrzesien_2017.pdf, dostęp: 13.01.2020

P. Talbot. "The Growing Impact Of AI On Marketing Strategy". Forbers, 2019 URL: https://www.forbes.com/sites/paultalbot/2019/09/21/the-growing-impact-of-ai-on-marketingstrategy/#7866346f7f14, dostęp 13.01.2020.

P. Tan, M. Steinbach, V. Kumar i in. "Cluster analysis: basic concepts and algorithms". W: Introduction to data mining, Addison-Wesley, 2006. str. 125-146.

S. C. Tan, K. M. Ting i T. F. Liu. "Fast anomaly detection for streaming data". W: Twenty-Second International Joint Conference on Artificial Intelligence, Barcelona, Spain, 07.2011.

Z. Tarapata, R. Kasprzyk i K. Banach. "Graph-network models and methods used to detect financial crimes with IAFEC graphs IT Tool". W: MATEC Web of Conferences. Vol. 210. EDP Sciences, 2018.

Y. Tkachenko. "Autonomous CRM control via CLV approximation with deep reinforcement learning in discrete and continuous action space", 2015, URL:https://arxiv.org/abs/1504.01840, dostęp 13.01.2020.

P. Torres i in. "An analysis of recurrent neural networks for botnet detection behavior". W: 2016 IEEE biennial congress of Argentina (ARGENCON). IEEE, 2016. str. 1-6.

A. Tuor, S. Kaplan, B. Hutchinson, N. Nichols i R. Sean. "Deep Learning for Unsupervised Insider Threat Detection in Structured Cybersecurity Data Streams". W: Workshops at the Thirty-First AAAI Conference on Artificial Intelligence, 02.2017.

A. Unni. "What You Need to Know About Crime-as-a-Service (CaaS)". 2018. URL: https://www.stickman.com.au/what-you-need-to-know-about-crime-as-a-service-caas/, dostęp 13.01.2020.

L. Van Der Maaten, E. Postma i J. Van Den Herik. "Dimensionality reduction: a comparative review". W: Journal of Machine Learning Research. 10.66-71. 01.2007. JMLR str. 13.

A. Vastel, P. Laperdrix, W. Rudametkin i R. Rouvoy. "FP-STALKER: Tracking browser fingerprint evolutions". W: 2018 IEEE Symposium on Security and Privacy (SP). IEEE, 05.2018. str. 728-741.

B. Venkatesh i J. Anuradha. "A Review of Feature Selection and Its Methods". W: Cybernetics and Information Technologies, 19.3.2019.

R. Vinayakumar, K. P. Soman i P. Poornachandran, "Long short-term memory based operation log anomaly detection". W: International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, 2017, str. 236-242.

R. Vinayakumar, K. P. Soman, K. K. S. Velan I S. Ganorkar. "Evaluating shallow and deep networks for ransomware detection and classification". W: International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, 2017, str. 259-265.

S. Vincenzi i in. "Application of a Random Forest algorithm to predict spatial distribution of the potential yield of Ruditapes philippinarum in the Venice lagoon, Italy". W: Ecological Modelling 222.8. Elsevier, 2011. str. 1471-1478.

M. Vlachos. "Dimensionality reduction. W: Encyclopedia of Machine Learning", 2010. str. 274-279.

D.-L. Vu, T.-K. Nguyen, T. V. Nguyen, T. N. Nguyen, F. Massacci, P. H. Phung "A Convolutional Transformation Network for Malware Classification". W: arXiv preprint arXiv:1909.07227, 2019.

C. JCH. Watkins i P. Dayan. "Q-learning". W: Machine learning 8.3-4. Springer US 1992. Str. 279 292.

W. Wei, B. Xi, i M. Kantarcioglu. "Adversarial Clustering: A Grid Based Clustering Algorithm Against Active Adversaries". W: arXiv preprint arXiv:1804.04780, 2018.

S. T. Wierzchoń, M. A. Kłopotek i I. Mika. "Algorytmy analizy skupień". Wydawnictwo WNT, 2015.

J. H. Wilson. "An analytical approach to detecting insurance fraud using logistic regression". W: Journal of Finance and accountancy. Academic and Business Research Institute. 2009. str. 1.

R.-S. Wu i Po-Hsuan Chou. "Customer segmentation of multiple category data in e-commerce using a soft-clustering approach". W: Electronic Commerce Research and Applications 10.32 Elsevier 2011. Str. 331-341.

T. Xiang i S. Gong. "Video behavior profiling for anomaly detection". W: IEEE transactions on pattern analysis and machine intelligence 30.5, IEEE, 2008. str. 893-908.

F. Xiao i in. "Malware Detection Based on Deep Learning of Behavior Graphs". W: Mathematical Problems in Engineering. Hindawi 2019, str. 1-10.

L. Xiong, S. Chitti, i L. Liu. "Mining multiple private databases using a knn classier". W: Proceedings of the 2007 ACM symposium on Applied computing. ACM, 2007. str. 435-440.

W. Xu i in. "Random rough subspace based neural network ensemble for insurance fraud detection". W: 2011 Fourth International Joint Conference on Computational Sciences and Optimization. IEEE, 2011. str. 1276-1280.

S. Yadav, V. Tiwari, B. Tiwari. "Privacy Preserving Data Mining With Abridge Time Using Vertical Partition Decision Tree". In Proceedings of the ACM Symposium on Women in Research 2016 (WIR '16). Association for Computing Machinery, New York, NY, USA, 158–164. 2016. DOI:https://doi.org/10.1145/2909067.2909097

X. Yan i J. Y. Zhang. "Early detection of cyber security threats using structured behavior modeling". W: ACM Transactions on Information and System Security 5, ACM, 2013.

Q. B. Yin, L. R. Shen, H. Q. Wang, "Intrusion detection based on hidden markov model". IEEE, Xi'an, China, 2003.

S. Yin i in. "Noisy training for deep neural networks in speech recognition". W: EURASIP Journal on Audio, Speech, and Music Processing 2015.1. SpringerOpen 2015. Str. 2.

D. Zakrzewska i J. Murlewski. "Clustering algorithms for bank customer segmentation". W: 5th International Conference on Intelligent Systems Design and Applications (ISDA'05). IEEE, 2005, str. 197-202.

O. B. Zhao. "Web Scraping". "Encyclopedia of Big Data 2017", Oregon State University, Springer International Publishing, 2017.

M. Zheng i in. "Cybersecurity research datasets: taxonomy and empirical analysis". W: 11th USENIX Workshop on Cyber Security Experimentation and Test 18. CSET 2018.